

Санкт-Петербургский государственный университет

Направление «Математическое обеспечение и администрирование  
информационных систем»

Профиль «Математические основы информатики»

Екатерина Андреевна Мальчевская

# Алгебраические байесовские сети: система логики-вероятностного вывода на основе матрично-векторных алгоритмов

Бакалаврская работа

Научный руководитель:  
проф. каф. инф., д. ф.-м. н., доц. Тулупьев А. Л.

Рецензент:  
доцент НИУ ВШЭ СПб, к.ф.-м.н. Сироткин А. В.

Санкт-Петербург  
2016

SAINT-PETERSBURG STATE UNIVERSITY

Mathematical Software and Information Systems Administration  
Mathematical foundations of computer science

Ekaterina Malchevskaia

Algebraic Bayesian networks:  
probabilistic-logic inference system based on  
matrix-vector algorithms

Bachelor's Thesis

Scientific supervisor:  
Professor, PhD, Associate Professor Tulupyev A. L.

Reviewer:  
Associate Professor, PhD Sirotkin A. V.

Saint-Petersburg  
2016

# Оглавление

<b>Введение</b>	<b>5</b>
<b>1. Алгебраическая байесовская сеть: автоматизация вывода</b>	<b>8</b>
1.1. Введение . . . . .	8
1.2. Виды логико-вероятностного вывода . . . . .	9
1.3. Библиотеки логико-вероятностного вывода . . . . .	12
1.4. Модернизация комплекса программ . . . . .	13
1.5. Выводы по главе . . . . .	13
<b>2. Теоретическая основа проекта</b>	<b>15</b>
2.1. Введение . . . . .	15
2.2. Модель ФЗ . . . . .	15
2.3. Алгоритмизация логико-вероятностного вывода . . . . .	20
2.4. Существующие программные реализации . . . . .	35
2.5. Выводы по главе . . . . .	38
<b>3. Архитектура разработанного программного комплекса</b>	<b>39</b>
3.1. Введение . . . . .	39
3.2. Особенности наследования компонентов в системе . . . . .	39
3.3. Структура создания и хранения ФЗ ABN . . . . .	42
3.4. Структуры локального-логико-вероятностного вывода и вспомогательные структуры . . . . .	43
3.5. Выводы по главе . . . . .	47
<b>4. Программная реализация</b>	<b>48</b>
4.1. Введение . . . . .	48
4.2. Примеры работы программного комплекса. Проверка и поддержание непротиворечивости . . . . .	48
4.3. Примеры работы программного комплекса. Априорный вывод . . . . .	50

4.4. Примеры работы программного комплекса. Апостериор- ный вывод . . . . .	53
4.5. Выводы по главе . . . . .	61
<b>Заключение</b>	<b>63</b>
<b>Список литературы</b>	<b>66</b>

# Введение

Актуальность темы. Одним из направлений современной информатики является обработка знаний с неопределенностью и изучение вероятностных графических моделей (ВГМ) [20, 14, 23]. Алгебраические байесовские сети (АБС) [8, 11] являются одним из классов ВГМ. Они представляют собой ненаправленные графы с идеалами конъюнктов в узлах. Конъюнкты, как и другие формулы, задаются над некоторым фиксированным алфавитом. При этом конъюнктам приписана скалярная или интервальная оценка вероятности истинности. Следуя [20, 23], будем называть идеалы конъюнктов с оценками вероятности фрагментами знаний (ФЗ).

Понятие алгебраических байесовских сетей было введено В.И. Городецким в 1993 году. С того момента теория существенно развилась, написаны работы, развивающие, уточняющие и дополняющие срез теории АБС, связанный со структурными представлениями, например первичными и вторичными структурами [2, 24]. Также были написаны работы, рассматривающие и развивающие подходы к логико-вероятностному выводу в АБС (поддержание непротиворечивости, априорный вывод, апостериорный вывод) [6, 5, 13, 4, 7].

Были разработаны и реализованы программные комплексы, базирующиеся на соответствующей теории. В 2009 году была разработана java-библиотека AlgBN Modeler j.v.01 [19] для работы с алгебраическими байесовскими сетями. Она позволяет хранить фрагменты знаний, изменять их, осуществлять возможные переходы от одних фрагментов знаний к другим. С помощью надстроек над этой библиотекой Algebraic Bayesian Networks Inferer и Algebraic Bayesian Networks Propagator [18, 17] поддерживаются непротиворечивость фрагмента знаний, локальный априорный и апостериорные выводы, то есть пропация детерминированных, стохастических и неточных свидетельств, а также некоторые виды глобального логико-вероятностного вывода. В 2011 году была разработана библиотека AlgBN KPB Reconciler cpp.v.01 [21] на C++, которая также реализует функциональность, необходимую для

работы с АБС, однако в данной ВКРб используются другие принципы построения структур классов и интерфейсов.

С одной стороны, по причине непрерывного развития и усовершенствования теории и появления матрично-векторных подходов к проведению логико-вероятностного вывода, с другой – из-за необходимости усовершенствования подходов, использовавшихся в уже имеющихся программных реализациях, возникла потребность в программном комплексе, который агрегирует ранее полученные результаты. В следствии чего было решено разработать библиотеку для локального логико-вероятностного вывода в АБС на C#, в рамках объемлющего проекта, с возможностью дальнейшей интеграции ее с другими разработками проекта, относящимися к структурному срезу теории АБС.

Объектом исследования являются алгебраические байесовские сети, а предметом исследования – локальный логико-вероятностный вывод в АБС с использованием матрично-векторных алгоритмов.

Целью данной выпускной квалификационной работы является автоматизация локального логико-вероятностного вывода в алгебраических байесовских сетях, включающего в себя проверку непротиворечивости предварительно созданного ФЗ, решение задачи локального априорного вывода и двух задач локального априорного вывода в АБС. Для достижения поставленной цели были сформированы задачи:

- 1) развить формализацию локального логико-вероятностного вывода для альтернативных моделей ФЗ;
- 2) реализация ФЗ и машин вывода на языке C#;
- 3) разработка примеров и документации.

Апробация результатов исследования. Результаты исследования были представлены на 1-ой Международной научной конференции «Интеллектуальные информационные технологии в технике и на производстве» и на Всероссийской научной конференции по проблемам информатики СПИСОК-2016.

Публикации. По теме выпускной квалификационной работы было подготовлено 5 публикаций, 4 из них были приняты к публикации.

Эта работа является частью более широких инициативных проектов, выполняющихся в лаборатории теоретических и междисциплинарных основ информатики СПИ-ИРАН под руководством А.Л. Тулупьева; кроме того, разработки были частично поддержаны грантами РФФИ 15-01-09001-а — «Комбинированный логико-вероятностный графический подход к представлению и обработке систем знаний с неопределенностью: алгебраические байесовские сети и родственные модели», 12-01-00945-а — «Развитие теории алгебраических байесовских сетей и родственных им логико-вероятностных графических моделей систем знаний с неопределенностью», 09-01-00861-а — «Методология построения интеллектуальных систем поддержки принятия решений на основе баз фрагментов знаний с вероятностной неопределенностью», а также проектом по ФЦНТП «Исследования и разработки по приоритетным направлениям развития науки и техники на 2002–2006 годы», 2006-РИ-19.0/001/211, Государственный контракт от «28» февраля 2006 г., № 02.442.11.7289, «Направленные циклы в байесовских сетях доверия: вероятностная семантика и алгоритмы логико-вероятностного вывода для программных комплексов с байесовской интеллектуальной компонентой».

# 1. Алгебраическая байесовская сеть: автоматизация вывода

## 1.1. Введение

Рассмотрим ситуацию, когда у нас есть несколько атомарных утверждений и есть эксперты, которые могут охарактеризовать взаимосвязи данных утверждений. Количеству утверждений соответствует экспоненциальное количество всех возможных взаимосвязей. Выходит, что при изначально большом количестве утверждений эксперт не сможет оценить все возможные взаимосвязи. Вероятностные графические модели и, в частности, алгебраические байесовские сети – одно из решений. Мы разбиваем множество всех утверждений на небольшие подмножества, называем их фрагментами знаний (ФЗ), и внутри них описываем все возможные взаимосвязи между элементами подмножества. Пример подобного ФЗ изображен на Рис. 1.

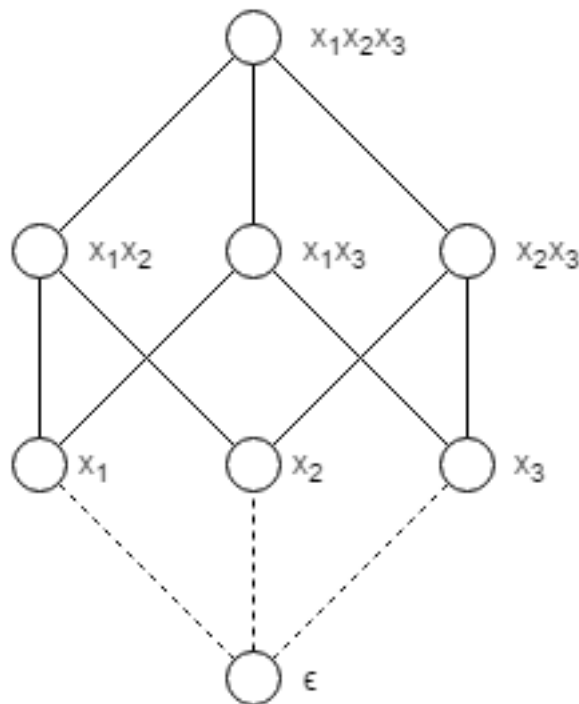


Рис. 1. Пример ФЗ построенного над  $A = \{x_1, x_2, x_3\}$ .

Множество всех получившихся ФЗ образуют базу фрагментов знаний (БФЗ). Изображение АБС таким образом, как это представлено на



Рис. 2, то есть рассмотрение сети, как множества ФЗ, мы называем первичной структурой АБС. Если мы рассмотрим граф смежности, в котором в качестве узлов будут выступать ФЗ, а ребро между двумя вершинами будет существовать, если у этих вершин, то есть ФЗ, есть общие атомы, то такой граф мы назовем вторичной структурой алгебраической байесовской сети. Примеры вторичной структуры алгебраической байесовской сети для АБС, представленной на Рис. 2, изображены на Рис. 3 и на Рис. 4.

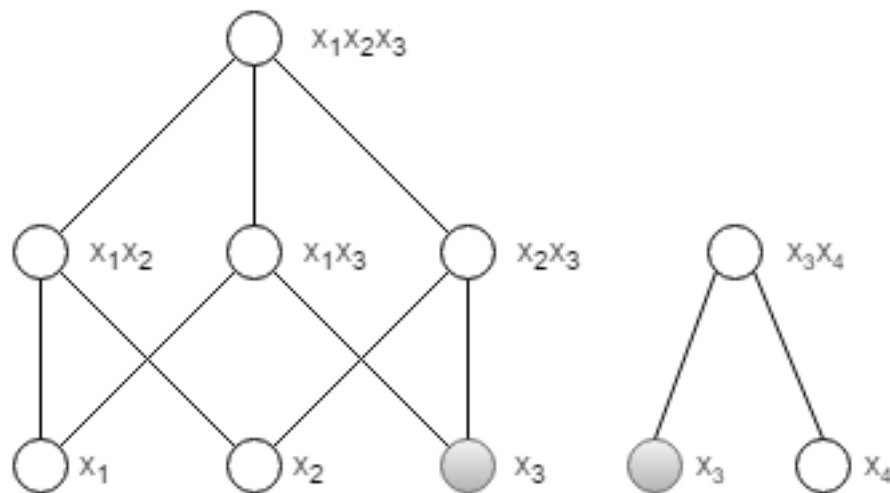


Рис. 2. Пример первичной структуры АБС, состоящей из двух ФЗ, построенной над  $A = \{x_1, x_2, x_3, x_4\}$ .

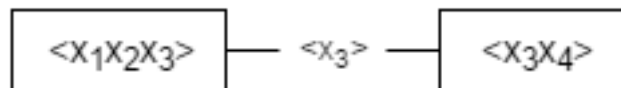


Рис. 3. Пример вторичной структуры АБС, состоящей из двух ФЗ, построенной над  $A = \{x_1, x_2, x_3, x_4\}$ .

Для определения степени неопределенности для утверждений и связей между ними мы можем использовать точечные(скалярные) оценки вероятности или же интервальные. На примере Рис. 5 можно увидеть, как могут быть заданы скалярные оценки вероятности.

## 1.2. Виды логико-вероятностного вывода

В качестве способа получения обусловленных данных, при условии уже имеющихся в АБС, существует аппарат логико-вероятностного вы-

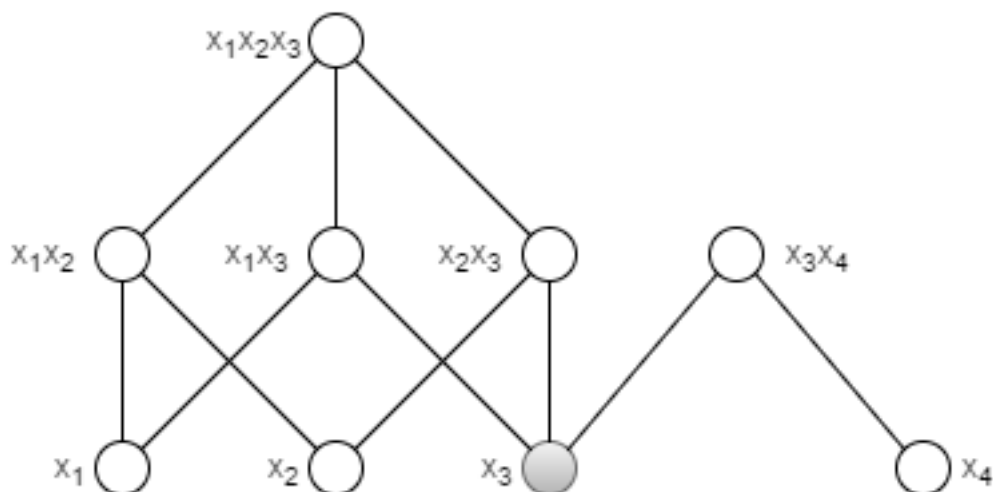


Рис. 4. Пример вторичной структуры АБС, состоящей из двух ФЗ, построенной над  $A = \{x_1, x_2, x_3, x_4\}$ .

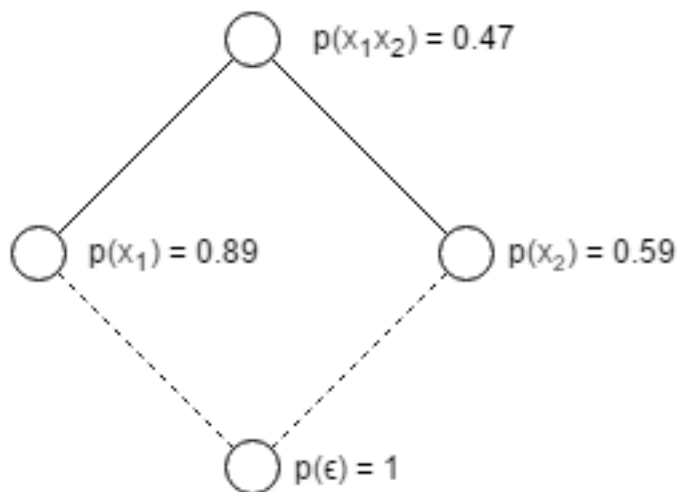


Рис. 5. Пример скалярных оценок вероятности заданных над ФЗ, построенного над  $A = \{x_1, x_2\}$ .

вода. Существует два вида логико-вероятностного вывода: локальный и глобальный. Локальный вывод рассматривается в пределах одного конкретного ФЗ, а глобальный затрагивает АБС в целом [8, 12]. В данной работе рассматривается в основном локальный вывод. Локальный логико-вероятностный вывод в АБС делится на априорный и апостериорный. Целью локального апостериорного логико-вероятностного вывода в АБС является решение двух задач. Разберем каждый вид вывода подробнее.

Априорный вывод заключается в том, что у нас есть ФЗ с заданными оценками вероятности и у нас есть пропозициональная форму-

ла, вероятность которой мы хотим вычислить, при условии заданных оценок. Алфавит, над которым задана пропозициональная формула, должен быть подмножеством алфавита, над которым задан рассматриваемый ФЗ.

Апостериорный вывод состоит в том, что нам приходит какая-то обуславливающая информация и при решении первой задачи апостериорного вывода мы считаем вероятность того, что такая информация могла прийти, при условии нашего ФЗ, а в случае решения второй задачи апостериорного вывода мы на основе пришедшего знания должны пересчитать оценки вероятностей нашего ФЗ.

Для проверки совместимости оценок вероятностей фрагментов знаний используются разные аппараты по проверке непротиворечивости (согласованности). На Рис. 6 приведен пример явно противоречивого ФЗ. Он оказывается противоречивым вследствие того, что вероятность атома не может быть меньше, чем вероятность конъюнкции, в которую входит данный атом. Также стоит отметить, что вероятность пустого конъюнкта, на изображениях обозначенного как  $\epsilon_{\wedge}$  равна 1, т.к пустой конъюнкт представляет собой тождественную истину в вероятностной логике, что не выполняется в примере, изображенном на Рис. 7.

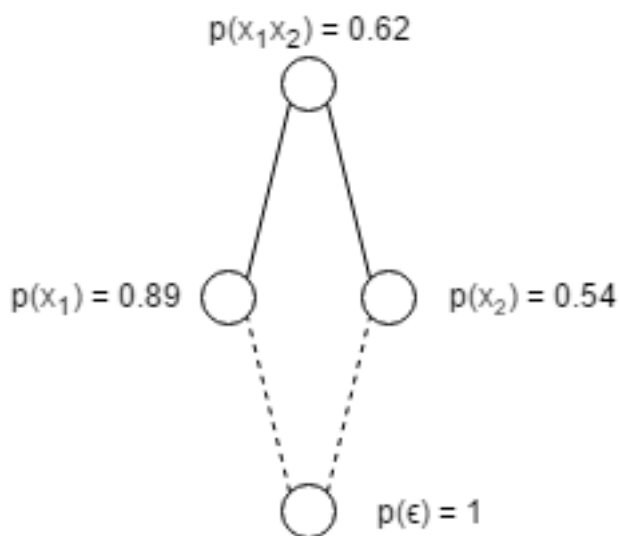


Рис. 6. Пример противоречивого ФЗ (1)

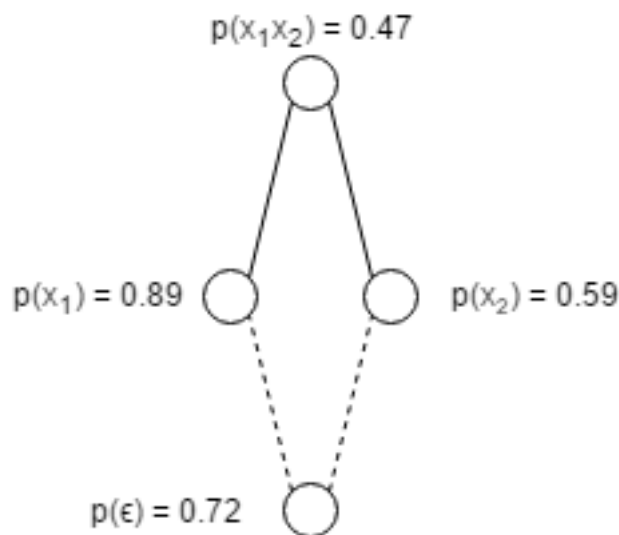


Рис. 7. Пример противоречивого ФЗ (2)

### 1.3. Библиотеки логико-вероятностного вывода

На настоящий момент известны библиотеки для работы с алгебраическими байесовскими сетями, а в частности две библиотеки, написанные на Java и на C++ [19, 21], которые включают в себя формирование ФЗ различных видов, их обработку, средства для локального априорного и апостериорного вывода и средства для глобального вывода.

Первая библиотека AlgBN Modeler j.v.01 была разработана на Java в 2009 году [19]. Особенностью данной библиотеки является разделение функциональности хранения фрагментов знаний с инструментами локального и глобального вывода, что позволяет назвать данную реализацию ФЗ «тонкой». При таком подходе, формирование и хранение ФЗ независимо от инструментов вывода, которые мы применяем.

Вторая библиотека AlgBN KPB Reconciler cpr.v.01 была разработана на C++ [21]. В реализации данной библиотеки применяется другой подход, который подразумевает объединение средств формирования и обработки фрагментов знаний с инструментами локального вывода. При таком подходе мы можем назвать реализацию ФЗ «толстой».

## 1.4. Модернизация комплекса программ

С 1980-х годов по настоящее время, были опубликованы монографии [8, 11, 20, 14, 23], диссертации [9, 3], статьи на тему алгебраических байесовских сетей и байесовских сетей в целом [16, 15]. Также развивалось и программное обеспечение и в связи с этим целью данной работы был перенос и усовершенствование частей библиотеки [19], отвечающих за формирование, хранение ФЗ, проведение локального логико-вероятностного вывода, на язык C# и платформу .NET.

Преимуществами реализации библиотеки именно на C# является то, что это позволит использовать библиотеки для решения задач линейного программирования и еще множество библиотек, реализованных на данной платформе. Также это позволит интегрировать полученные результаты с другой частью библиотеки, связанной с первичными, вторичными и более сложными структурами АБС.

В разработанной библиотеке основное внимание уделено обеспечению использования особенностей представления бинарных ФЗ как скалярных и интервальных, а скалярных как интервальных и на то, чтобы реализация соответствовала «тонкой».

Целью данной выпускной квалификационной работы является автоматизация локального логико-вероятностного вывода в алгебраических байесовских сетях, включающего в себя проверку непротиворечивости предварительно созданного ФЗ, решение задачи локального априорного вывода и двух задач локального априорного вывода в АБС. Для достижения поставленной цели были сформированы задачи:

1. развить формализацию локального логико-вероятностного вывода для альтернативных моделей ФЗ;
2. реализация ФЗ и машин вывода на языке C#;
3. разработка примеров и документации.

## 1.5. Выводы по главе

В данной главе рассмотрены основные понятия, связанные с тео-

рией АБС и локальным логико-вероятностным выводом и его видами. Упомянуты некоторые существующие программные реализации. Приведена цель и задачи данной выпускной квалификационной работы.

## 2. Теоретическая основа проекта

### 2.1. Введение

В данной главе введем необходимые математические понятия и определения, которые формально описывают модель ФЗ в алгебраической байесовской сети и те операции, которые над ней можно выполнять. Рассмотрим базовые принципы и алгоритмы по работе с локальными априорным и апостериорным логико-вероятностными выводами. Раскроем и разовьем понятия, относящиеся к логико-вероятностному выводу. В основу дальнейшего изложения взят ряд источников [9, 3, 14, 7].

### 2.2. Модель ФЗ

#### 2.2.1. ФЗ, построенные над идеалом конъюнктов

Алгебраическая байесовская сеть является одной из моделей, которые представляют данные с неопределенностью, а также вероятностной графической моделью. Неопределенность возникает вследствие того, что конъюнктам в узлах фрагментов знаний, из которых состоит АБС, приписывается некоторая вероятность, степень неточности.

**Определение 2.1** [3] *Алфавит  $A = \{x_i\}_{i=0}^{n-1}$  – конечное множество атомарных пропозициональных формул (атомов). Введем нумерацию атомов с нуля.*

Теперь над атомами из указанного алфавита определим наборы пропозициональных формул.

**Определение 2.2** [3] *Конъюнкт (цепочка конъюнкций) – это конъюнкция некоторого числа атомарных переменных вида:*

$$x_{i_1} \wedge x_{i_2} \wedge \dots \wedge x_{i_k}.$$

Мы можем такие формулы записывать, опустив знаки конъюнций. Не будем их различать, если это не будет вызывать неоднозначность:

$$x_{i_1} x_{i_2} \dots x_{i_k}.$$

**Определение 2.3** [3] Идеал конъюнктов (идеал цепочек конъюнкций) – это множество вида

$$\{x_{i_1} \wedge x_{i_2} \wedge \cdots \wedge x_{i_k} \mid 0 \leq i_1 < i_2 < \cdots < i_k \leq n-1, 0 \leq k \leq n\}.$$

Обозначим идеал конъюнктов над алфавитом  $A$ , как  $C_A$ .

**Определение 2.4** [3] Каждому конъюнкту  $x_{i_1}x_{i_2} \cdots x_{i_k}$  сопоставим число

$$2^{i_1} + 2^{i_2} + \cdots + 2^{i_k}.$$

– индекс конъюнкта.

**Определение 2.5** [3] Литерал(аргументное место)  $\tilde{x}_i$  обозначает, что на его месте в формуле может стоять либо  $x_i$ , либо  $\bar{x}_i$ .

**Определение 2.6** [3] Квант над алфавитом  $A = \{x_i\}_{i=0}^{n-1}$  – это конъюнкция, которая для любого атома алфавита содержит либо этот атом, либо его отрицание.

**Определение 2.7** [3] Множество квантов над алфавитом  $A = \{x_i\}_{i=0}^{n-1}$ :

$$Q = \{\tilde{x}_{i_1}\tilde{x}_{i_2} \cdots \tilde{x}_{i_k}\}.$$

Для нумерации квантов будем пользоваться методом, аналогичным нумерации конъюнктов, более подробно о нем можно прочитать в [3].

После введения нумерации можно определить векторы оценок вероятности квантов и конъюнктов соответственно [3]:

$$\mathbf{P}_q = \begin{pmatrix} p(q_0) \\ p(q_1) \\ \vdots \\ p(q_{2^n-1}) \end{pmatrix}$$



и

$$\mathbf{P}_c = \begin{pmatrix} 1 \\ p(c_1) \\ \vdots \\ p(c_{2^n-1}) \end{pmatrix}.$$

Теперь введем некоторые определения, связанные с видами фрагментов знаний.

**Определение 2.8** [3] Фрагмент знаний ( $\Phi Z$ ) со скалярными оценками – это пара вида  $(C, p)$ , где  $C$  – идеал конъюнктов, а  $p$  – функция из  $C$  в интервал  $[0; 1]$

**Определение 2.9** [3] Фрагмент знаний ( $\Phi Z$ ) с интервальными оценками – это структура вида  $(C, p)$ , где  $C$  – идеал конъюнктов, а  $p$  – функция из  $C$  в множество интервалов вида

$$\{[a; b] : a, b \in [0; 1], a \leq b\}.$$

Введем базовые определения непротиворечивости и согласуемости  $\Phi Z$  [3].

**Определение 2.10** [3] Пусть задан  $\Phi Z$  со скалярными оценками вероятности  $(C, p)$ . Мы говорим, что он непротиворечив(согласован), и обозначаем это как  $\text{Consistent}[(C, p)]$  тогда и только тогда, когда существует вероятность  $p_F$ , заданная над множеством пропозициональных формул  $F(A)$  (где  $A$  – это алфавит, над которым построен  $\Phi Z$ ), такая что

$$\forall c \in C \ p_F = p(c).$$

**Определение 2.11** [3] Пусть задан  $\Phi Z$  с интервальными оценками вероятности  $(C, p)$ . Мы говорим, что он непротиворечив(согласован), и обозначаем это как  $\text{Consistent}[(C, p)]$  тогда и только тогда, когда для любого конъюнкта  $c \in C$  и любого  $\epsilon \in p(c)$  найдется функция  $p_{c, \epsilon} : C \rightarrow [0, 1]$  такая, что  $p_{c, \epsilon} = \epsilon$  и  $(C, p_{c, \epsilon})$  – непротиворечивый в смысле предыдущего определения.

**Определение 2.12** [3] Пусть задан ФЗ с интервальными оценками вероятностей  $(C, p)$ . Мы говорим, что он согласуем тогда и только тогда, когда существует непротиворечивый ФЗ с интервальными оценками  $(C, p')$  такой, что

$$\forall c \in C \ p'(c) \subseteq p(c).$$

Введем основные понятия локального апостериорного логико-вероятностного вывода в рамках рассматриваемой модели ФЗ. Для формализации локального логико-вероятностного вывода нам необходимо ввести определение математической модели свидетельства в АБС.

**Определение 2.13** [3] Задача локального априорного вывода состоит в том, чтобы на основе непротиворечивого фрагмента знаний построить оценки истинности пропозициональной формулы, заданной над тем же алфавитом, что и фрагмент знаний.

**Определение 2.14** [3] Свидетельство – новые «обуславливающие» данные, которые поступили во ФЗ, и с учетом которых нам требуется пересмотреть все (или только некоторые) оценки. Для обозначения свидетельства будут использоваться угловые скобки –  $\langle \dots \rangle$ .

Далее оговорим, что из себя представляют задачи апостериорного логико-вероятностного вывода.

**Определение 2.15** [3] Первая задача апостериорного вывода состоит в том, чтобы оценить вероятность истинности свидетельства при уже заданных оценках вероятности истинности элементов ФЗ.

**Определение 2.16** [3] Вторая задача апостериорного вывода состоит в том, чтобы оценить условные вероятности истинности элементов ФЗ при предположении, что имеет место быть свидетельство.

### 2.2.2. Альтернативные математические модели ФЗ

Вслед за [14] определим две альтернативные математические модели, одна из которых образована над идеалам дизъюнктов, другая – над множеством квантов.

**Определение 2.17** [14] *Фрагментом знаний над идеалом дизъюнктов  $C_d$  со скалярными оценками вероятности определяется как упорядоченная пара  $(C_v, p)$ , где  $C_v$  – идеал дизъюнктов, лежащий в основе ФЗ, а  $p$  – функция на  $C_v$ , задающая скалярные (точечные) оценки вероятностей:*

$$p : C_v \leftarrow [0; 1].$$

Значения этой функции, упорядоченные по аналогии с конъюнктами, записываются в виде вектора  $\mathbf{D}$  или  $\mathbf{P}_d$ . Компоненты вектора нумеруются с 0 до  $2^m - 1$ . Обратим внимание, что нулевой компонент вектора  $D_0 = 0$ , так как пустой дизъюнкт соответствует тождественной лжи.

**Определение 2.18** [14] *ФЗ над множеством квантов  $C_q$  со скалярными оценками вероятности определяется как упорядоченная пара  $(Q, p)$ , где  $Q$  – множество квантов, лежащее в основе ФЗ, а  $p$  – функция на  $Q$ , задающая скалярные (точечные) оценки вероятностей:*

$$p : Q \leftarrow [0; 1].$$

Значения этой функции, упорядоченные по аналогии с конъюнктами, записываются в виде вектора  $\mathbf{Q}$  или  $\mathbf{P}_q$ . Компоненты вектора нумеруются с 0 до  $2^m - 1$ .

Используя введенные векторные обозначения ФЗ над идеалом дизъюнктов, можно рассмотреть как пару  $(C_v, \mathbf{D})$ , состоящую из множества квантов и вектора скалярных оценок этих дизъюнктов, а ФЗ над множеством квантов как пару  $(Q, \mathbf{Q})$ , состоящую из множества квантов и вектора скалярных оценок этих квантов.

После введения нумерации можем определить вектор оценок вероятностей дизъюнктов соответственно.

$$\mathbf{P}_d = \begin{pmatrix} 0 \\ p(d_1) \\ \vdots \\ p(d_{2^n-1}) \end{pmatrix}.$$

## 2.3. Алгоритмизация логико-вероятностного вывода

Прежде чем приступить непосредственно к рассмотрению алгоритмов решения задач локального логико-вероятностного вывода, разберем аппарат, осуществляющий проверку непротиворечивости вероятностных оценок ФЗ и некоторые понятия с ним связанные.

### 2.3.1. Поддержание непротиворечивости

Рассмотрим вероятность на ранее введенном множестве всех квантов  $Q$ . Запишем требующиеся ограничения в векторном виде:

$$\begin{aligned} \mathbf{P}_q &\geq \mathbf{0}, \\ (\mathbf{1}, \mathbf{P}_q) &= 1. \end{aligned}$$

Вектор  $\mathbf{1}$  – вектор подходящей размерности, состоящий из единиц. В большинстве случаев, размерность вектора  $2^n$ .

Матрица  $\mathbf{I}_n$ , упомянутая ранее, имеет чёткую структуру [3], которую можно описать рекуррентно.

$$\mathbf{I}_1 = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}, \dots, \mathbf{I}_n = \mathbf{I}_1 \otimes \mathbf{I}_{n-1} = \mathbf{I}_1 \otimes \mathbf{I}_1^{[n-1]} = \mathbf{I}_1^{[n]}.$$

При этом выполняется соотношение:

$$\mathbf{P}_q = \mathbf{I}_n \times \mathbf{P}_c.$$

Здесь  $\otimes$  кронекерово произведение матриц. Также используется об-

ратная матрице  $\mathbf{I}_n$  матрица  $\mathbf{J}_n$ , которая удовлетворяет условию:

$$\mathbf{P}_c = \mathbf{J}_n \times \mathbf{P}_q,$$

$$\mathbf{J}_1 = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \mathbf{J}_n = \mathbf{J}_1 \otimes \mathbf{J}_{n-1} = \mathbf{J}_1 \otimes \mathbf{J}_1^{[n-1]} = \mathbf{J}_1^{[n]}.$$

Теперь, вслед за [3], сформулируем определения непротиворечивости фрагментов знаний, описанные ранее на матрично-векторном языке.

**Определение 2.19** Пусть задан ФЗ со скалярными оценками вероятностей  $(C, \mathbf{P}_c)$ . мы говорим, что он непротиворечив тогда и только тогда, когда

$$\mathbf{I}_n \times \mathbf{P}_c \geq \mathbf{0}.$$

**Определение 2.20** Пусть задан ФЗ с интервальными оценками вероятностей  $(C, \mathbf{P}_c^-, \mathbf{P}_c^+)$ . Мы говорим, что он непротиворечив тогда и только тогда, когда

$$\forall i : 1 \leq i \leq 2^n - 1 \quad \forall \epsilon : \mathbf{P}_c^-[i] \leq \epsilon \leq \mathbf{P}_c^+[i]$$

$$\exists \mathbf{P}_c : (\mathbf{P}_c^- \leq \mathbf{P}_c \leq \mathbf{P}_c^+) \ \& \ (\mathbf{P}_c[i] = \epsilon) \ \& \ (\mathbf{I}_n \times \mathbf{P}_c \geq \mathbf{0}).$$

**Определение 2.21** Пусть задан ФЗ с интервальными оценками вероятностей  $(C, \mathbf{P}_c^-, \mathbf{P}_c^+)$ . Мы говорим, что он согласуем тогда и только тогда, когда существует непротиворечивый ФЗ с интервальными оценками  $(C, \mathbf{P}^-, \mathbf{P}^+)$ , такой что  $\mathbf{P}_c^- \leq \mathbf{P}^-$  и  $\mathbf{P}^+ \leq \mathbf{P}_c^+$ .

Обоснование приведенных выше рассуждений, дополнительные факты, утверждения и теоремы и примеры приведены в [3].

Далее рассмотрим зависимости, связанные с поддержанием непротиворечивости фрагментов знаний, представленными идеалом дизъюнктов и множеством квантов.

Введем обозначения: вектор  $\mathbf{1}_m$  – вектор, состоящий из  $2^m$  единиц и вектор  $\mathbf{0}$  – вектор, состоящий из того же числа элементов, причем все будут нулевыми, кроме самого нижнего, который будет равен единице.

Определим вектор  $\overline{\mathbf{P}_c}$  как вектор, содержащий вероятность каждого конъюнкта, составленного из отрицаний атомов.

Имеют место соотношения:

$$\begin{aligned}\mathbf{1}_m - \overline{\mathbf{P}_c} &= \mathbf{D}, \\ \mathbf{1}_m - \mathbf{D} &= \overline{\mathbf{P}_c}.\end{aligned}$$

**Определение 2.22** Для ФЗ  $C_d = (C_v, \mathbf{D})$  задание скалярных оценок считается непротиворечивым, если

$$\mathbf{ID} \leq \mathbf{0}.$$

**Определение 2.23** Для ФЗ  $C_q = (Q, \mathbf{Q})$  задание скалярных оценок считается непротиворечивым, если

$$\mathbf{Q} \geq \mathbf{0},$$

$$(\mathbf{Q}, \mathbf{1}) = 1.$$

Доказательства и обоснования приведенных фактов и закономерностей представлено в [14].

### 2.3.2. Априорный вывод

Рассмотрим локальный априорный вывод над двумя видами ФЗ: скалярным и интервальным.

**Априорный вывод над скалярным ФЗ.** Опишем решение задачи априорного вывода для случая ФЗ со скалярными оценками вероятности. Рассмотрим непротиворечивый ФЗ со скалярными оценками  $(C, \mathbf{P}_c)$ . Мы можем выразить вероятность произвольной пропозициональной формулы через вероятности квантов, входящих в её СДНФ.

**Определение 2.24** Пусть  $f$  – произвольная пропозициональная формула, тогда характеристическим вектором формулы  $f$  мы будем называть вектор  $\chi_f$ , состоящий из  $2^n$  элементов, такой, что

$$\chi_f[i] = \begin{cases} 0, & \text{если } q_i \notin S_f; \\ 1, & \text{если } q_i \in S_f \end{cases}$$

, где  $q_i$  обозначает  $i$ -й квант, а  $S_f$  – множество квантов, содержащихся в СДНФ  $\phi$ -лы  $f$ .

Рассмотрим пропозициональную формулу  $f$ , истинность которой требуется оценить. На основе прежде рассмотренных знаний мы можем сделать вывод, что

$$p(f) = (\chi_f, \mathbf{P}_q).$$

Используя ранее рассмотренные формулы, мы можем перейти от вектора с вероятностями квантов к вектору с вероятностями конъюнктов и получим

$$p(f) = (\chi_f, \mathbf{I}_n \times \mathbf{P}_c) = (\mathbf{I}_n^T \times \chi_f, \mathbf{P}_c).$$

Обозначим  $\mathbf{L}_f = \mathbf{I}_n^T \times \chi_f$ , тогда

$$p(f) = (\mathbf{L}_f, \mathbf{P}_c).$$

**Априорный вывод над интервальным ФЗ.** Перейдем к ФЗ с интервальными оценками вероятности. В данном случае однозначно определить вероятность истинности пропозициональной формулы не удастся, но можно найти максимальную и минимальную оценки:

$$p^-(f) = \min_{\mathfrak{D} \cup \mathfrak{E}}(\mathbf{L}_f, \mathbf{P}_c), \quad p^+(f) = \max_{\mathfrak{D} \cup \mathfrak{E}}(\mathbf{L}_f, \mathbf{P}_c).$$

### 2.3.3. Апостериорный вывод

В теории АБС рассматриваются три вида свидетельств: детерминированные, стохастические и неточные.

**Определение 2.25** *Детерминированное свидетельство – некоторое означивание цепочки конъюнкций, рассматриваемое в качестве поступившего свидетельства. Данное свидетельство обозначается  $\langle c_i, c_j \rangle$ . Здесь  $c_i, c_j$  – конъюнкты, состоящие из атомов, получивших положительные и отрицательные означивания соответственно.*

**Определение 2.26** *Стохастическое свидетельство – непротиворечивый ФЗ, заданный над  $C'$  – подыдеале  $C$ , со скалярными оценками, которые определяет вероятности истинности элементов соответствующего подыдеала. Данное свидетельство обозначается  $\langle (C', \mathbf{P}_c) \rangle$ .*

**Определение 2.27** *Неточное свидетельство – непротиворечивый ФЗ с  $C'$  – подыдеале  $C$  интервальными оценками, которые определяет вероятности истинности элементов соответствующего подыдеала. Данное свидетельство обозначается  $\langle (C', \mathbf{P}_c^-, \mathbf{P}_c^+) \rangle$ .*

Все рассмотренные выше задачи образуют локальный логико-вероятностный вывод в АБС, который был реализован в программном комплексе. Также большинство задач ЛВВ сводится к решению задач линейного программирования (ЛП), то есть поиску минимума и максимума некоторой целевой функции, при заданных ограничениях, что позволяет использовать известные методы решения задач ЛП.

**Детерминированное свидетельство и ФЗ со скалярными оценками.** Для начала рассмотрим процесс обработки детерминированного свидетельства  $\langle i, j \rangle$ , поступившего в ФЗ со скалярными оценками  $(C, \mathbf{P}_c)$ .

**Определение 2.28** [3] *Введем дополнительные обозначения.*

$$\mathbf{H}^+ = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix},$$

$$\mathbf{H}^- = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix},$$



$$\mathbf{H}^o = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}.$$

А теперь определим матрицу, как тензорное произведение:

$$\mathbf{H}^{\langle i,j \rangle} = \overline{\mathbf{H}}_{n-1}^{\langle i,j \rangle} \otimes \overline{\mathbf{H}}_{n-2}^{\langle i,j \rangle} \otimes \cdots \otimes \overline{\mathbf{H}}_0^{\langle i,j \rangle},$$

где

$$\overline{\mathbf{H}}_k^{\langle i,j \rangle} = \begin{cases} \mathbf{H}^+, & \text{если } x_k \text{ входит в } c_i; \\ \mathbf{H}^-, & \text{если } x_k \text{ входит в } c_j; \\ \mathbf{H}^o, & \text{иначе.} \end{cases}$$

**Утверждение 2.3.1** [3] Пусть задан ФЗ со скалярными оценками  $(C, \mathbf{P}_c)$ , тогда вероятность истинности свидетельства  $\langle i, j \rangle$  при условии заданных ФЗ может быть вычислена по формуле

$$p(\langle i, j \rangle | (C, \mathbf{P}_c)) = (\mathbf{1}, \mathbf{H}^{\langle i,j \rangle} \times \mathbf{P}_q).$$

Подробное обоснование данного факта можно найти в [3].

**Определение 2.29** [3] Пусть имеется свидетельство  $\langle c_i, c_j \rangle$ , а  $c_j$  представляет собой конъюнкт  $x_{j_1} x_{j_2} \cdots x_{j_k}$ , тогда этому свидетельству соответствует пропозициональная формула

$$q_{\langle i,j \rangle} = c_i \overline{x}_{j_1} \overline{x}_{j_2} \cdots \overline{x}_{j_k}.$$

Далее рассмотрим вектор, состоящий из условных вероятностей квантов при заданном свидетельстве:

$$\mathbf{P}_q^{\langle i,j \rangle} = \begin{pmatrix} p(q_0 | q_{\langle i,j \rangle}) \\ \vdots \\ p(q_{2^n-1} | q_{\langle i,j \rangle}) \end{pmatrix}.$$

Теперь воспользуемся определением матрицы  $H$ . Вышерассмотренный вектор можно записать в матрично-векторном виде:

$$\mathbf{P}_q^{(i,j)} p(\langle c_i, c_j \rangle) = \mathbf{H}^{(i,j)} \times \mathbf{P}_q.$$

По аналогии построим вектор:

$$\mathbf{P}_c^{(i,j)} = \begin{pmatrix} p(c_0 | q_{\langle i,j \rangle}) \\ \vdots \\ p(c_{2^n-1} | q_{\langle i,j \rangle}) \end{pmatrix}.$$

**Утверждение 2.3.2** [3] Векторы  $\mathbf{P}_c^{(i,j)}$  и  $\mathbf{P}_q^{(i,j)}$  связаны соотношениями

$$\begin{aligned} \mathbf{P}_c^{(i,j)} &= \mathbf{J}_n \times \mathbf{P}_q^{(i,j)}, \\ \mathbf{P}_q^{(i,j)} &= \mathbf{I}_n \times \mathbf{P}_c^{(i,j)}. \end{aligned}$$

Доказательство данного утверждения также приведено в [3].

После ряда преобразований получим :

$$\mathbf{P}_q^{(i,j)} p(\langle c_i, c_j \rangle) = \mathbf{J}_n \times \mathbf{H}^{(i,j)} \times \mathbf{I}_n \times \mathbf{P}_c.$$

Рассмотрим матрицу, которая при фиксированных  $i$  и  $j$  будет задаваться однозначно:

$$\mathbf{T}^{(i,j)} = \mathbf{J}_n \times \mathbf{H}^{(i,j)} \times \mathbf{I}_n.$$

Подставим эту матрицу в ранее рассмотренное выражение и получим:

$$\mathbf{P}_q^{(i,j)} p(\langle c_i, c_j \rangle) = \mathbf{T}^{(i,j)} \times \mathbf{P}_c.$$

**Теорема 2.1** [3] В случае ФЗ со скалярными оценками решением первой задачи апостериорного вывода для свидетельства  $\langle i, j \rangle$  является  $(\mathbf{T}^{(i,j)} \times \mathbf{P}_c)[0]$  .

**Теорема 2.2** [3] Если  $p(\langle c_i, c_j \rangle) \neq 0$ , то решение второй задачи апостериорного вывода в случае детерминированного свидетельства даёт формула:

$$\mathbf{P}_q^{\langle i,j \rangle} = \frac{1}{(\mathbf{T}^{\langle i,j \rangle} \times \mathbf{P}_c)[0]} \mathbf{T}^{\langle i,j \rangle} \times \mathbf{P}_c = \frac{\mathbf{T}^{\langle i,j \rangle} \times \mathbf{P}_c}{(\mathbf{T}^{\langle i,j \rangle} \times \mathbf{P}_c)[0]}.$$

Если  $p(\langle c_i, c_j \rangle) = 0$ , то поступившее свидетельство имело вероятность 0, то есть произойти не могло, в этом случае будем говорить, что свидетельство невероятное.

**Детерминированное свидетельство и ФЗ с интервальными оценками.** Теперь рассмотрим ФЗ с интервальными оценками вероятности  $\mathbf{P}_c^-$  и  $\mathbf{P}_c^+$ . В этом случае решение первой и второй задач апостериорного вывода для детерминированного свидетельства можно представить как решение серии задач линейного программирования (ЛП).

Рассмотрим первую задачу апостериорного вывода.

**Утверждение 2.3.3** [3] Для детерминированного свидетельства  $\langle i, j \rangle$  существует вектор  $\mathcal{L}^{\langle i,j \rangle}$  такой, что для решения первой задачи апостериорного вывода при заданном свидетельстве и интервальных оценках во фрагменте знаний необходимо решить две задачи линейного программирования, соответствующие поиску величин:

$$\begin{aligned} \min_{\mathcal{D} \cup \mathcal{E}} \{(\mathcal{L}^{\langle i,j \rangle}, \mathbf{P}_c)\}, \\ \max_{\mathcal{D} \cup \mathcal{E}} \{(\mathcal{L}^{\langle i,j \rangle}, \mathbf{P}_c)\}. \end{aligned}$$

Перейдем ко второй задаче апостериорного вывода. Её решением будут следующие величины:

$$\begin{aligned} \min_{\mathcal{D} \cup \mathcal{E}} \{\mathbf{P}_c^{\langle i,j \rangle}[i]\}, \\ \max_{\mathcal{D} \cup \mathcal{E}} \{\mathbf{P}_c^{\langle i,j \rangle}[i]\}. \end{aligned}$$

**Теорема 2.3** [3] Решением второй задачи локального апостериорного вывода в случае детерминированного свидетельства и ФЗ с интер-

важными оценками является решение серии задач ЛП по поиску:

$$\begin{aligned} \min\{\mathbf{T}^{(i,j)} \times \mathbf{D}\}, \\ \max\{\mathbf{T}^{(i,j)} \times \mathbf{D}\}, \end{aligned}$$

при условиях:

$$\{\lambda \mathbf{P}_c^- \leq \mathbf{D}\} \cup \{\lambda \mathbf{P}_c^+ \geq \mathbf{D}\} \cup \{\mathbf{I}_n \times \mathbf{D} \geq \mathbf{0}\} \cup \{(\mathbf{L}^{(i,j)}, \mathbf{D}) = 1\} \cup \{\lambda \geq 0\}.$$

Доказательства обоих вышерассмотренных утверждений приведены в [3].

### Стохастические свидетельства.

**Определение 2.30** [3] Пусть задан идеал конъюнктов  $C$ , тогда будем использовать обозначение  $A_c$  для сужения алфавита  $A$  до элементов, над которыми задан  $C$ , а  $Q_c$ , для множества квантов, заданных над этим алфавитом.

Пусть нам задан ФЗ со скалярными оценками  $(C, \mathbf{P}_c)$ . И, согласно определению стохастического свидетельства, задан непротиворечивый ФЗ  $(C', \mathbf{P}_c^a)$ , такой, что  $C' \subseteq C$ . Так как  $(C', \mathbf{P}_c^a)$  – непротиворечивый, то на основе оценок вероятностей для конъюнктов  $\mathbf{P}_c^a$ , можно построить оценки вероятности для квантов  $Q_{C'} - \mathbf{P}_q^a = \mathbf{I} \times \mathbf{P}_c^a$ . Элементы множества  $Q_{C'}$  являются взаимоисключающими, то есть при любом зафиксированном означивании всех атомов истинным будет только один и только один элемент множества  $Q_{C'}$ . Кроме того, каждый элемент  $Q_{C'}$  можно трактовать как соответствующее детерминированное свидетельство, для ФЗ  $(C, \mathbf{P}_c)$ . Пользуясь этими свойствами мы можем формализовать стохастическое свидетельство следующим образом: свидетельство  $(C', \mathbf{P}_c^a)$  – случайный элемент, принимающий значение детерминированных свидетельств, соответствующих элементам  $Q_{C'}$  с вероятностями  $\mathbf{P}_q^a$ . При такой формализации мы можем перейти к описанию двух задач апостериорного вывода. Так как свидетельство является случайным элементом, а в качестве решений первой и второй

задач апостериорного вывода нужно получить численные значения, то решениями будем считать соответствующие математические ожидания.

Рассмотрим решение первой задачи апостериорного вывода. Для детерминированного свидетельства  $\langle i, j \rangle$  это  $(\mathbf{T}^{\langle i, j \rangle} \times \mathbf{P}_c)[0]$  Нам необходимо уметь сопоставлять индексы множества  $Q_{C'}$  индексам детерминированных свидетельств. Для этого определим вспомогательную функцию  $\text{GInd}(i, m)$ , где  $m$  – это индекс наибольшего элемента  $C'$  в алфавите  $A$ ,  $i$  – индекс конъюнкта в алфавите  $A_{C'}$ , а функция возвращает индекс соответствующего конъюнкта в алфавите  $A$ .

Обозначим мощность алфавита  $A_{C'}$  через  $n'$ . Выпишем в этих обозначениях математическое ожидание, которое и будет решением первой задачи апостериорного вывода:

$$\sum_{i=0}^{2^{n'}-1} (\mathbf{T}^{\langle \text{GInd}(i, m), \text{GInd}(2^{n'}-1-i, m) \rangle} \times \mathbf{P}_c)[0] (\mathbf{I} \times \mathbf{P}_c^a)[i].$$

Соответственно, если все детерминированные свидетельства возможны, то решением второй задачи апостериорного вывода для стохастического свидетельства будет:

$$\mathbf{P}_c^{\langle (C', \mathbf{P}_c^a) \rangle} = \sum_{i=0}^{2^{n'}-1} \frac{\mathbf{T}^{\langle \text{GInd}(i, m), \text{GInd}(2^{n'}-1-i, m) \rangle} \times \mathbf{P}_c}{(\mathbf{T}^{\langle \text{GInd}(i, m), \text{GInd}(2^{n'}-1-i, m) \rangle} \times \mathbf{P}_c)[0]} (\mathbf{I} \times \mathbf{P}_c^a)[i].$$

Если какое-либо детерминированное свидетельство невозможно и при этом оценки, задаваемые стохастическим свидетельством для данного детерминированного, нулю не равны, то мы говорим, что такое свидетельство невероятное.

Теперь рассмотрим ФЗ с интервальными оценками вероятности и стохастическое свидетельство. Для решения первой задачи апостериорного вывода нам нужно найти максимум и минимум функции:

$$\sum_{i=0}^{2^{n'}-1} (\mathbf{T}^{\langle \text{GInd}(i, m), \text{GInd}(2^{n'}-1-i, m) \rangle} \times \mathbf{P}_c)[0] (\mathbf{I} \times \mathbf{P}_c^a)[i].$$

Так как функция – линейная, то нам необходимо решить две соответствующие задачи ЛП.

Для второй задачи апостериорного вывода функция перестает быть линейной. Из-за этого мы не сможем точно найти соответствующие максимумы и минимумы, не выходя за рамки задач линейного программирования. Будем строить не точное решение, а накрывающее его. Для этого вычислим:

$$\mathbf{P}_c^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle, \min} = \min \left( \frac{\mathbf{T}^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle} \times \mathbf{P}_c}{(\mathbf{T}^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle} \times \mathbf{P}_c)[0]} \right)$$

и

$$\mathbf{P}_c^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle, \max} = \max \left( \frac{\mathbf{T}^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle} \times \mathbf{P}_c}{(\mathbf{T}^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle} \times \mathbf{P}_c)[0]} \right).$$

Будем считать решением второй задачи апостериорного вывода в случае стохастического свидетельства и интервальных оценок оценки:

$$\sum_{i=0}^{2^{n'}-1} \mathbf{P}_c^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle, \min} (\mathbf{I} \times \mathbf{P}_c^a)[i]$$

и

$$\sum_{i=0}^{2^{n'}-1} \mathbf{P}_c^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle, \max} (\mathbf{I} \times \mathbf{P}_c^a)[i].$$

**Неточные свидетельства.** Рассмотрим последний вид свидетельств – неточные.

Неточное свидетельство представляет собой фрагмент знаний с интервальными оценками  $(C', \mathbf{P}_c^{a,-}, \mathbf{P}_c^{a,+})$ . Будем трактовать его следующим образом [3]: неточное свидетельство  $(C', \mathbf{P}_c^{a,-}, \mathbf{P}_c^{a,+})$ , это семейство всевозможных стохастических свидетельств  $(C', \mathbf{P}_c^a)$ , таких что  $\mathbf{P}_c^{a,-} \leq \mathbf{P}_c^a \leq \mathbf{P}_c^{a,+}$ .

В данном случае мы сталкиваемся с выходом из задач ЛП уже на этапе решения первой задачи апостериорного вывода. Приближим ре-

шение, зафиксировав максимумы и минимумы значений оценок для каждого детерминированного. Получим две линейные оценки, которые накроют истинные интервальные оценки для решения первой задачи апостериорного вывода:

$$\begin{aligned} \min & \left( \sum_{i=0}^{2^{n'}-1} (\mathbf{T}^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle} \times \mathbf{P}_c)[0](\mathbf{I} \times \mathbf{P}_c^a)[i] \right), \\ \max & \left( \sum_{i=0}^{2^{n'}-1} (\mathbf{T}^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle} \times \mathbf{P}_c)[0](\mathbf{I} \times \mathbf{P}_c^a)[i] \right). \end{aligned}$$

В случае, когда исходный ФЗ имеет скалярные оценки, эти формулы точные, а не накрывающие.

Для решения второй задачи апостериорного вывода достаточно найти:

$$\begin{aligned} \min & \left( \sum_{i=0}^{2^{n'}-1} \mathbf{P}_c^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle, \min}(\mathbf{I} \times \mathbf{P}_c^a)[i] \right), \\ \max & \left( \sum_{i=0}^{2^{n'}-1} \mathbf{P}_c^{\langle \text{GInd}(i,m), \text{GInd}(2^{n'}-1-i,m) \rangle, \max}(\mathbf{I} \times \mathbf{P}_c^a)[i] \right). \end{aligned}$$

**Поддержание непротиворечивости и апостериорный вывод в ФЗ над идеалом дизъюнктов.** Запишем пример вектора вероятностей элементов идеала дизъюнктов фрагмента знаний, построенного над алфавитом мощности 2 из атомов  $x_1, x_2$ , и вектор, получающийся при вычитании его из единичного вектора.

$$\mathbf{P}_d = \begin{pmatrix} 0 \\ p(x_1) \\ p(x_2) \\ p(x_1 \vee x_2) \end{pmatrix}, \mathbf{1} - \mathbf{P}_d = \begin{pmatrix} 1 \\ p(\bar{x}_1) \\ p(\bar{x}_2) \\ p(\bar{x}_1 \bar{x}_2) \end{pmatrix}.$$

Рассмотрим соотношение для перехода от вектора оценок вероятностей дизъюнктов к вектору оценок вероятностей квантов  $\bar{\mathbf{E}}_n \mathbf{I}_n (\mathbf{1} - \mathbf{P}_d) =$

$\mathbf{P}_q$ , где  $\bar{\mathbf{E}}_n = \begin{pmatrix} 0 & \dots & 1 \\ \dots & 1 & \dots \\ 1 & \dots & 0 \end{pmatrix}$  – матрица, в которой на всех позициях кроме второй диагонали стоят нули, а вторая диагональ занята единицами. Равенство становится очевидным при раскрытии левой части уравнения.

Домножим обе части уравнения на матрицу  $\mathbf{J}_n$ , получим:  $\mathbf{J}_n \bar{\mathbf{E}}_n \mathbf{I}_n (\mathbf{1} - \mathbf{P}_d) = \mathbf{J}_n \mathbf{P}_q = \mathbf{P}_c$ . Введем новую матрицу  $\mathbf{K}_n$ :

$$\begin{aligned} \mathbf{K}_n = \mathbf{J}_n \bar{\mathbf{E}}_n \mathbf{I}_n &= \left( \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}^{[n]} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}^{[n]} \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}^{[n]} \right) = \\ &= \left( \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \right)^{[n]} = \\ &= \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix}^{[n]}, \mathbf{K}_1 = \mathbf{K}. \end{aligned}$$

Используя новое обозначение мы можем записать выражение вектора вероятностей идеала конъюнктов через вектор вероятности идеала дизъюнктов:  $\mathbf{K}_n (\mathbf{1} - \mathbf{P}_d) = \mathbf{P}_c$ . Также заметим, что  $\mathbf{K}_n^{-1} = \mathbf{K}_n$ . Введем обозначение:  $\mathbf{P}_d' = \mathbf{1} - \mathbf{P}_d$ . Перепишем ранее полученные уравнения используя введенное обозначение:

$$\mathbf{P}_c = \mathbf{K}_n \mathbf{P}_d'.$$

Рассмотрим уравнения апостериорного вывода для ФЗ над идеалом дизъюнктов, используя полученные ранее формулы для идеала конъюнктов.

Следующие теоремы были сформулированы в соавторстве с А.А. Золотиным. Математическая постановка – А.А. Золотин, формальное доказательство – Е.А. Мальчевская.

#### Теорема 2.4 [1]

$$\begin{aligned} p(\langle c_i, c_j \rangle) &= (\mathbf{d}^{\langle i, j \rangle}, \mathbf{P}_d'), \\ \text{где } \mathbf{d}^{\langle i, j \rangle} &= \bigotimes_{k=0}^{k=n-1} \tilde{\mathbf{d}}_k^{\langle i, j \rangle}, \end{aligned}$$



$$\tilde{\mathbf{d}}_k^{\langle i,j \rangle} = \begin{cases} \mathbf{d}^+, & \text{если } x_k \text{ входит в } c_i; \\ \mathbf{d}^-, & \text{если } x_k \text{ входит в } c_j; \\ \mathbf{d}^o, & \text{иначе.} \end{cases}$$

$$\text{причем } \mathbf{d}^+ = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \mathbf{d}^- = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \mathbf{d}^o = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \text{ и } \mathbf{P}_{\mathbf{d}'} = \mathbf{1} - \mathbf{P}_{\mathbf{d}}.$$

**Доказательство.** Выполним ряд преобразований. Вероятность свидетельства, поступившего в ФЗ над идеалом конъюнктов равна  $p(\langle i, j \rangle) = (\mathbf{r}^{\langle i,j \rangle}, \mathbf{P}_{\mathbf{c}})$  [22]. Подставим вместо  $\mathbf{P}_{\mathbf{c}}$  полученное ранее выражение для вектора идеала дизъюнктов:

$$\begin{aligned} p(\langle i, j \rangle) &= (\mathbf{r}^{\langle i,j \rangle}, \mathbf{P}_{\mathbf{c}}) = (\mathbf{r}^{\langle i,j \rangle}, \mathbf{K}_n \mathbf{P}_{\mathbf{d}'} ) = (\mathbf{K}_n^T \mathbf{r}^{\langle i,j \rangle}, \mathbf{P}_{\mathbf{d}'} ), \\ \mathbf{K}_n^T &= \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix}^{[n]}, \mathbf{r}^{\langle i,j \rangle} = \bigotimes_{k=0}^{k=n-1} \tilde{\mathbf{r}}_k^{\langle i,j \rangle}, \\ \tilde{\mathbf{r}}_k^{\langle i,j \rangle} &= \begin{cases} \mathbf{r}^+, & \text{если } x_k \text{ входит в } c_i; \\ \mathbf{r}^-, & \text{если } x_k \text{ входит в } c_j; \\ \mathbf{r}^o, & \text{иначе.} \end{cases} \\ , \mathbf{r}^+ &= \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \mathbf{r}^- = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \mathbf{r}^o = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \\ (\mathbf{K}_n^T \mathbf{r}^{\langle i,j \rangle}, \mathbf{P}_{\mathbf{d}'} ) &= ((\mathbf{K}^T)^{[n]} (\bigotimes_{k=0}^{k=n-1} \tilde{\mathbf{r}}_k^{\langle i,j \rangle}), \mathbf{P}_{\mathbf{d}'} ) = (\bigotimes_{k=0}^{k=n-1} (\mathbf{K}^T \tilde{\mathbf{r}}_k^{\langle i,j \rangle}), \mathbf{P}_{\mathbf{d}'} ). \end{aligned}$$

Заметим, что вектор  $\tilde{\mathbf{r}}_k^{\langle i,j \rangle}$  может принимать только три значения, а матрица  $\mathbf{K}_n^T$  – одно, следовательно будет существовать только 3 возможных означивания вектора  $\tilde{\mathbf{d}}_k^{\langle i,j \rangle} = \mathbf{K}^T \tilde{\mathbf{r}}_k^{\langle i,j \rangle}$ . Рассмотрим их:

$$\begin{aligned} \mathbf{d}^+ &= \mathbf{K}^T \mathbf{r}^+ = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}, \\ \mathbf{d}^- &= \mathbf{K}^T \mathbf{r}^- = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \\ \mathbf{d}^o &= \mathbf{K}^T \mathbf{r}^o = \begin{pmatrix} 1 & 1 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \end{aligned}$$

Тогда, зная, что  $\mathbf{d}^{\langle i,j \rangle} = \bigotimes_{k=0}^{k=n-1} \tilde{\mathbf{d}}_k^{\langle i,j \rangle}$  получаем, что  $p(\langle c_i, c_j \rangle) = (\mathbf{d}^{\langle i,j \rangle}, \mathbf{P}_{\mathbf{d}'} )$ .

■

**Теорема 2.5** [1]

$$\mathbf{P}_d'^{\langle i,j \rangle} = \frac{1}{(\mathbf{d}^{\langle i,j \rangle}, \mathbf{P}_d')} \mathbf{M}^{\langle i,j \rangle} \times \mathbf{P}_d',$$

$$\text{где } \mathbf{M}^{\langle i,j \rangle} = \bigotimes_{k=0}^{k=n-1} \tilde{\mathbf{M}}_k^{\langle i,j \rangle},$$

$$\tilde{\mathbf{M}}_k^{\langle i,j \rangle} = \begin{cases} \mathbf{M}^+, & \text{если } x_k \text{ входит в } c_i; \\ \mathbf{M}^-, & \text{если } x_k \text{ входит в } c_j; \\ \mathbf{M}^o, & \text{иначе.} \end{cases}$$

$$\text{причем } \mathbf{M}^+ = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}, \mathbf{M}^- = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}, \mathbf{M}^o = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \text{ и } \mathbf{P}_d' = \mathbf{1} - \mathbf{P}_d.$$

**Доказательство.**

Решение второй задачи апостериорного вывода для ФЗ над идеалом конъюнктов записывается как  $\mathbf{P}_c^{\langle i,j \rangle} = \frac{1}{(\mathbf{d}^{\langle i,j \rangle}, \mathbf{P}_d')} \mathbf{T}^{\langle i,j \rangle} \times \mathbf{P}_c$ . Заметим, что  $\mathbf{P}_c^{\langle i,j \rangle} = \mathbf{K}_n(1 - \mathbf{P}_d^{\langle i,j \rangle})$ . Подставим вместо  $\mathbf{P}_c$  и  $\mathbf{P}_c^{\langle i,j \rangle}$  полученные ранее выражения для векторов идеалов дизъюнктов, а знаменатель заменим на выражение из предыдущей теоремы:

$$\begin{aligned} \mathbf{P}_c^{\langle i,j \rangle} &= \frac{1}{(\mathbf{r}^{\langle i,j \rangle}, \mathbf{P}_c)} \mathbf{T}^{\langle i,j \rangle} \times \mathbf{P}_c \\ \mathbf{K}_n(1 - \mathbf{P}_d^{\langle i,j \rangle}) &= \frac{1}{(\mathbf{d}^{\langle i,j \rangle}, \mathbf{P}_d')} \mathbf{T}^{\langle i,j \rangle} \times \mathbf{K}_n \times (1 - \mathbf{P}_d) \\ \mathbf{P}_d'^{\langle i,j \rangle} &= \frac{1}{(\mathbf{d}^{\langle i,j \rangle}, \mathbf{P}_d')} \mathbf{K}_n^{-1} \times \mathbf{T}^{\langle i,j \rangle} \times \mathbf{K}_n \times (1 - \mathbf{P}_d) \end{aligned}$$

Подробнее распишем произведение матриц и воспользуемся тем, что  $\mathbf{K}_n^{-1} = \mathbf{K}_n$ :

$$\mathbf{K}_n^{-1} \times \mathbf{T}^{\langle i,j \rangle} \times \mathbf{K}_n = \mathbf{K}_n \times \mathbf{T}^{\langle i,j \rangle} \times \mathbf{K}_n = (\mathbf{K})^{[n]} \times \mathbf{T}^{\langle i,j \rangle} \times (\mathbf{K})^{[n]}$$

Ранее было получено [23], что  $\mathbf{T}^{\langle i,j \rangle} = \bigotimes_{k=0}^{k=n-1} \tilde{\mathbf{T}}_k^{\langle i,j \rangle}$ ,

$$\tilde{\mathbf{T}}_k^{\langle i,j \rangle} = \begin{cases} \mathbf{T}^+, & \text{если } x_k \text{ входит в } c_i; \\ \mathbf{T}^-, & \text{если } x_k \text{ входит в } c_j; \\ \mathbf{T}^o, & \text{иначе.} \end{cases}$$

$$\text{причем } \mathbf{T}^+ = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}, \mathbf{T}^- = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}, \mathbf{T}^o = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Так как матрица  $\tilde{\mathbf{T}}_k^{\langle i,j \rangle}$ , как было показано, может принимать 3 зна-

чения, а матрица  $\mathbf{K}$  – одно, то у матрицы  $\tilde{\mathbf{M}}_k^{\langle i,j \rangle} = \mathbf{K} \times \tilde{\mathbf{T}}_k^{\langle i,j \rangle} \times \mathbf{K}_n$  есть 3 варианта означивания:

$$\mathbf{M}^+ = \mathbf{K}\mathbf{T}^+\mathbf{K} = \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix},$$

$$\mathbf{M}^- = \mathbf{K}\mathbf{T}^-\mathbf{K} = \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix},$$

$$\mathbf{M}^o = \mathbf{K}\mathbf{T}^o\mathbf{K} = \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 1 & -1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Получим, что  $\mathbf{P}_d'^{\langle i,j \rangle} = \frac{1}{(\mathbf{d}^{\langle i,j \rangle}, \mathbf{P}_d')} \mathbf{M}^{\langle i,j \rangle} \times \mathbf{P}_d'$ ,

и  $\mathbf{M}^{\langle i,j \rangle} = \bigotimes_{k=0}^{k=n-1} \tilde{\mathbf{M}}_k^{\langle i,j \rangle}$ ,

$$\tilde{\mathbf{M}}_k^{\langle i,j \rangle} = \begin{cases} \mathbf{M}^+, & \text{если } x_k \text{ входит в } c_i; \\ \mathbf{M}^-, & \text{если } x_k \text{ входит в } c_j; \\ \mathbf{M}^o, & \text{иначе.} \end{cases}$$

причем  $\mathbf{M}^+ = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}$ ,  $\mathbf{M}^- = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}$ ,  $\mathbf{M}^o = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ . ■

## 2.4. Существующие программные реализации

В качестве основы использовалась библиотека AlgBN KPB Reconciler src.v.01 [21], упомянутая ранее и написанная в 2011 году на C++. Фрагмент знаний в случае данной библиотеки представляется одним классом со всей информацией, такой как вероятностные оценки и размер фрагмента знаний, хранимой в полях. Методы, отвечающие за формирование ФЗ, такие как конструкторы, и методы, отвечающие за локальный апприорный вывод и апостериорный вывод, также хранятся в этом классе. Такой клиент называется «толстым», потому что реализация различных функциональностей не разделена по разным классам, а находится в одном, объемном классе. Преимущество данного подхода в том, что всевозможные виды ФЗ, такие как интервальные, скалярные и бинарные, мы можем представить с помощью единой структуры, то есть унифицировать работу со всеми видами фрагментов знаний.

Также была взята за основу библиотека AlgBN Modeler j.v.01 [19],

которая была написана в 2009 году. Основной её особенностью и обоснованием того, что клиент – «тонкий», является то, что в ней разные функциональности разнесены в разные структуры классов, таким образом каждая подструктура библиотеки отвечает за свою задачу.

Однако для интеграции со структурной частью объемлющего проекта было решено переработать существующие решения на языке C#. В данной библиотеке основной упор сделан на структуру интерфейсов. В работе рассматриваются 3 вида фрагментов знаний: интервальный, скалярный и бинарный. В частности, было замечено, что бинарный фрагмент может быть задан, как скалярный, а скалярный, соответственно, как интервальный. Из данного наблюдения получилось сразу две цепи наследования. Первая цепь это наследование `getter`'ов, методов, которые считывают вероятностные оценки фрагментов знаний. Бинарный ФЗ может быть считан, как скалярный, а скалярный как интервальный. На Рис. 8, который был заимствован из [9], показано, как происходит наследование.

Вторая цепь наследования это наследование `setter`'ов, методов, которые задают вероятностные оценки фрагментов знаний. Бинарный ФЗ может быть задан как скалярный, а скалярный как интервальный. На Рис. 8 показано, как в данном случае происходит наследование.

Вышеупомянутые знания были применены и к фрагментам знаний, которые формально представляются, как идеал конъюнктов, дизъюнктов или квантов. В следствии чего были получены взаимосвязи, которые можно видеть на Рис. 8. Также упомянутые зависимости были использованы и в структурах пропагаторов и инфереров, то есть машин осуществляющих апостериорный и априорный вывод соответственно.

В программном комплексе использовалась библиотека для решения задач линейного программирования `lp_solve55`, которая позволяет задавать целевую функцию, ограничения на переменные, максимизировать или минимизировать целевую функцию.

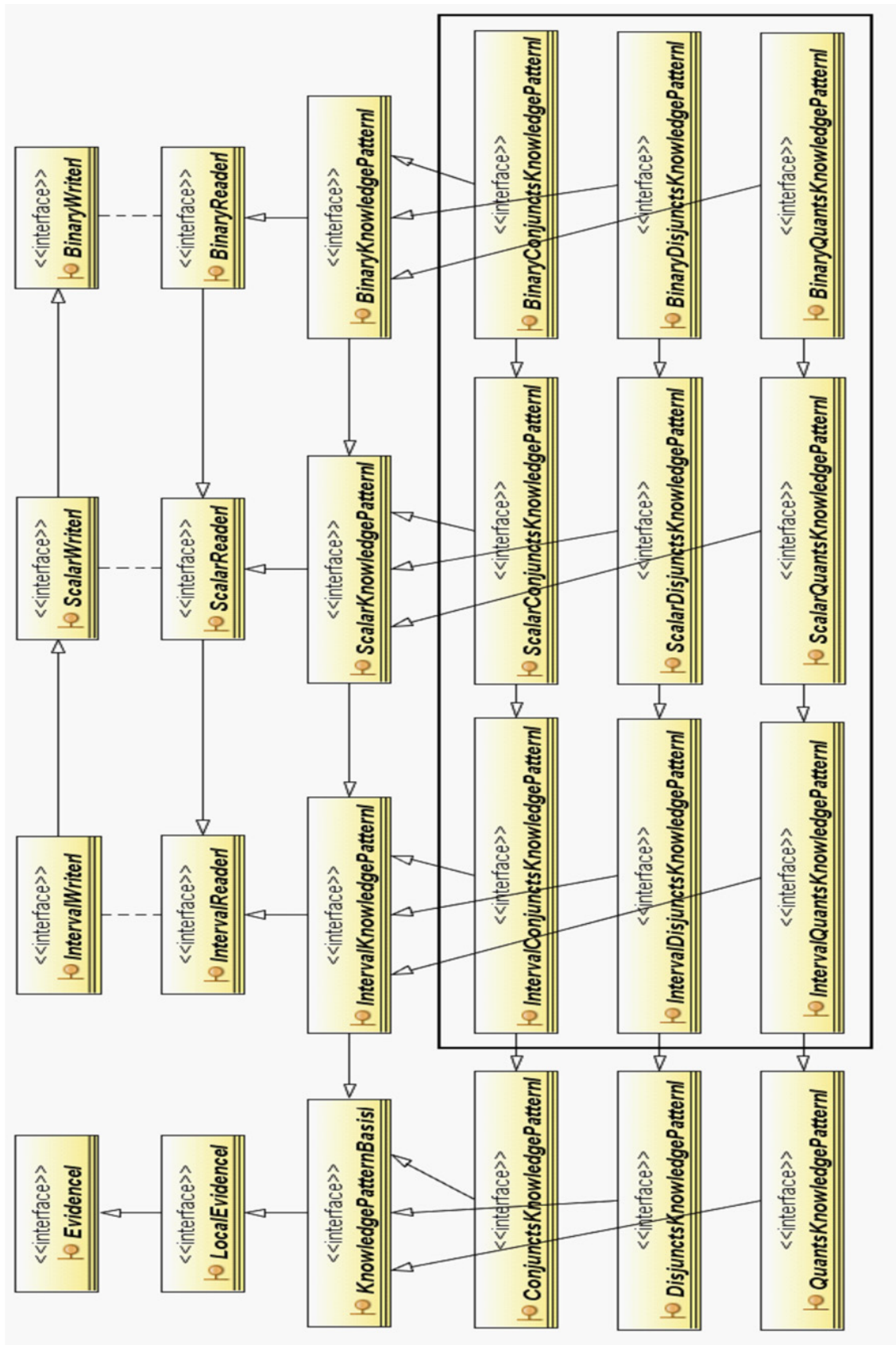


Рис. 8. Иерархия интерфейсов [19].

## 2.5. Выводы по главе

В данной главе описаны как классическая модель фрагмента знаний, так и альтернативные, связанные с этими моделями определения. Рассмотрены математические аппараты для осуществления локального логико-вероятностного вывода: априорного и апостериорного. Доказаны две теоремы, в которых формализуется апостериорный вывод в ФЗ, построенных над идеалом дизъюнктов. Описаны существующие программные реализации, которые были взяты за основу, а также упомянута библиотека, которая использовалась в разработанном программном комплексе.

## **3. Архитектура разработанного программного комплекса**

### **3.1. Введение**

В предыдущей главе была рассмотрена теоретическая основа работы, в этой главе рассмотрим некоторые аспекты программной реализации. В начале опишем структуру ABN, состоящую из интерфейсов и классов, которые ответственны за хранение и создание фрагментов знаний с интервальными, скалярными и бинарными оценками вероятности. Приведем срез структуры, связанный с интервальными ФЗ, для бинарных и скалярных ФЗ набор и функциональность классов и интерфейсов похожая. Разберем структуру Inferrer, отвечающую за поддержание и проверку непротиворечивости и за априорный вывод. Приведем только та часть структуры, которая работает со скалярными ФЗ, в интервальном случае набор классов и интерфейсов аналогичен. Приведем структуру Propagator, состоящую из классов, абстрактных классов и интерфейсов, отвечающую за апостериорный логико-вероятностный вывод, а точнее за пропагацию детерминированного, стохастического или неточного свидетельства во ФЗ со скалярными или интервальными оценками вероятности. А также опишем вспомогательные классы.

### **3.2. Особенности наследования компонентов в системе**

Архитектура данной работы воспроизводит архитектуру библиотеки AlgBN Modeler j.v.01 [19]. На Рис. 9 изображена диаграмма связи всех интерфейсов, которые используются в структуре данной библиотеки, отвечающей за создание и хранение ФЗ (ABN).

Также была прослежена зависимость между абстрактными классами, которая изображена на Рис. 10.

Мнемонику абстрактного класса KnowledgePatternBasis [19] было решено изменить на KPSSupp (сокращенно от Knowledge Pattern Support),

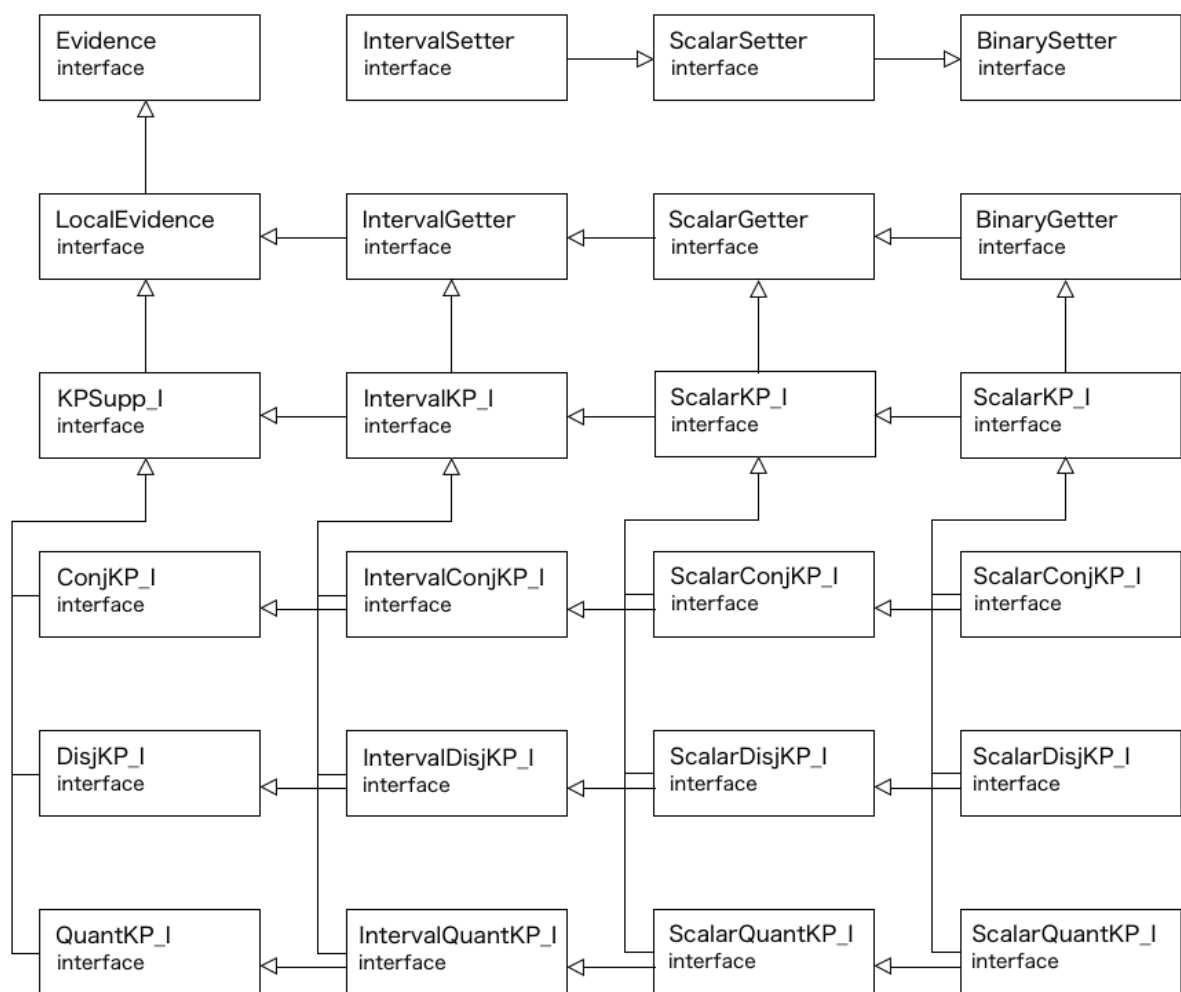


Рис. 9. Диаграмма интерфейсов.

так как это существенно.

И далее были реализованы такие классы, как IntervalDisjKP, IntervalConjKP, IntervalQuantKP, ScalarDisjKP, ScalarConjKP, ScalarQuantKP, BinaryDisjKP, BinaryConjKP, BinaryQuantKP, связь между которыми изображена на Рис. 11. В некоторых местах код был упрощен по причине особенности платформы, на которой была написана программная часть, но это все же требует дополнительных исследований.

Особенности наследования интерфейсов и классов, отвечающих за хранение фрагментов знаний, использованы и в принципах наследования в других структурах программного комплекса, отвечающих за априорный и апостериорный логико-вероятностный вывод, которые мы



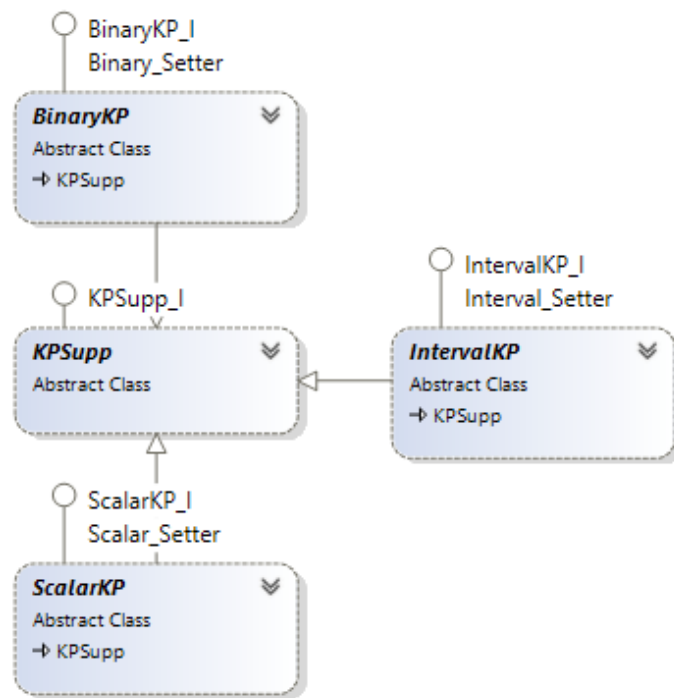


Рис. 10. Диаграмма зависимости абстрактных классов.

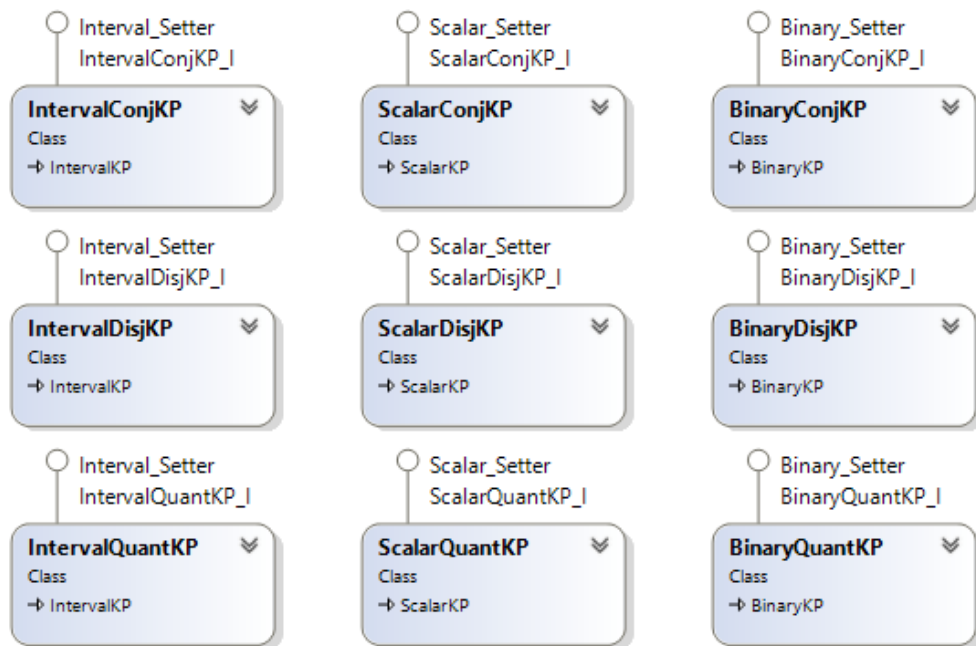


Рис. 11. Диаграмма классов.

разберем далее.

### 3.3. Структура создания и хранения ФЗ АВН

На Рис. 12 приведена диаграмма интерфейсов структуры АВН, которая отвечает за интервальный фрагмент знаний.

Evidence и LocalEvidence позволяют всем классам, наследующих у них, считывать глобальный индекс ФЗ.

KPSupp\_I содержит в себе методы, которые считывают мощность алфавита, на котором построен рассматриваемый ФЗ, и количество в нем элементов.

ConjKP\_I, DisjKP\_I, QuantKP\_I показывают принадлежность рассматриваемого ФЗ к какому-либо классу представления ФЗ: конъюнктов, дизъюнктов или квантов соответственно.

IntervalGetter и IntervalSetter содержат методы, которые считывают вероятностные оценки ФЗ и задают их соответственно.

И IntervalConjKP\_I, IntervalDisjKP\_I, IntervalQuantKP\_I обозначают методы, присущие интервальным конъюнктивному, дизъюнктивному и квантовому ФЗ соответственно.

Структуры LinearForm и Extention позволяют для каждого ФЗ задавать список пропозициональных формул, для которых в последствии будет выполняться априорный вывод. Пропозициональная формула задается вектором из коэффициентов в разложении формулы в СНДФ по конъюнктам исходного ФЗ.

На Рис. 13 показана взаимосвязь классов IntervalConjKP, IntervalDisjKP, IntervalQuantKP, которые реализуют методы тех интерфейсов, которые были рассмотрены ранее и также наследуют часть реализованных методов абстрактного класса IntervalKP.

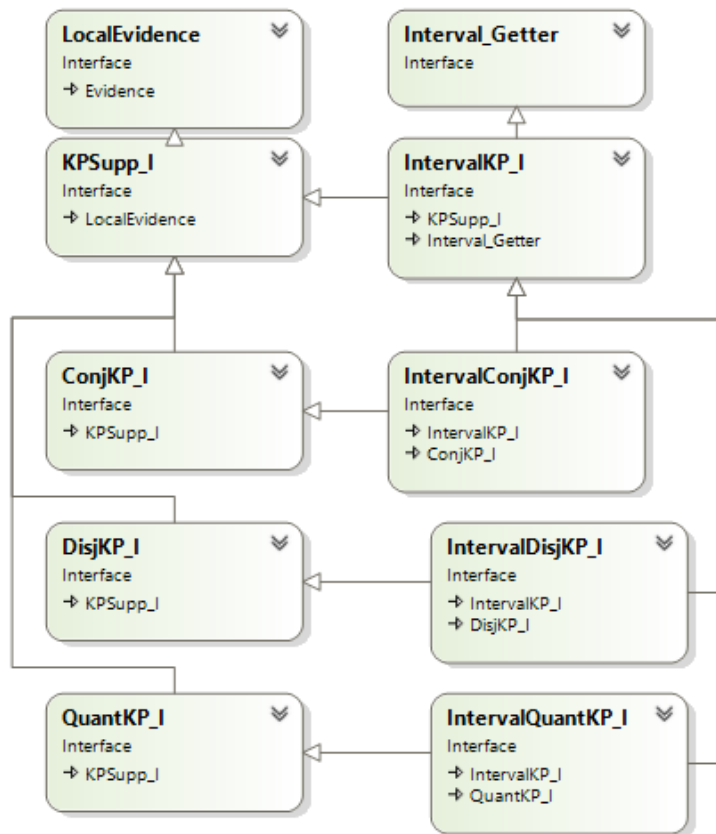


Рис. 12. Диаграмма интерфейсов.

### 3.4. Структуры локального-логико-вероятностного вывода и вспомогательные структуры

#### 3.4.1. Структура Inferrer априорного логико-вероятностного вывода

Интерфейс InferrerI содержит методы, которые выдают сведения о совместимости оценок и проверяют совместимы ли оценки ФЗ и улучшились они. Оценки могут улучшиться только у ФЗ с интервальными оценками. То есть реализующий класс для данного интерфейса должен обрабатывать ФЗ, проверять, противоречив ли ФЗ или нет и проверять, стали ли оценки более точными.

С помощью методов интерфейса LocalInferrerI можно проверить непротиворечивость ФЗ и получить копию ФЗ, заданного на входе, с переобозначенными оценками, если оценки уточнились.

Метод интерфейса ScalarLocalInferrerI либо возвращает ФЗ, постро-

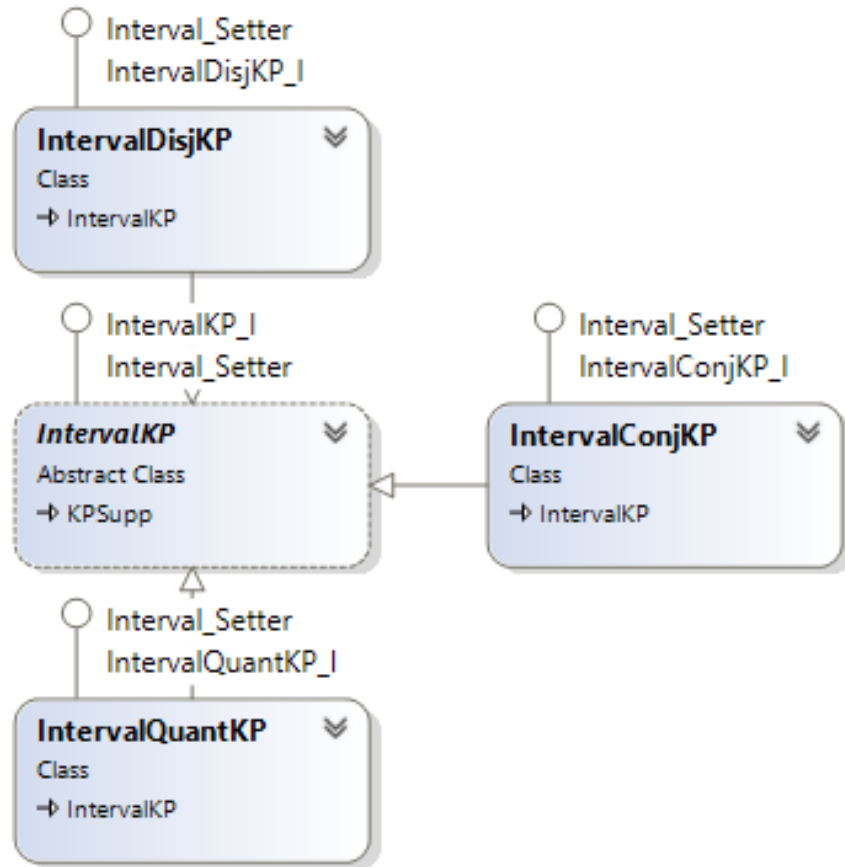


Рис. 13. Диаграмма классов, относящихся к интервальному ФЗ.

енный над квантами, соответствующий исходному ФЗ, либо сообщает о том, что исходный ФЗ противоречив.

Класс `ScalarConjunctsLocalInferer` реализует методы всех вышеупомянутых интерфейсов, а также содержит средства, в частности метод `processKP()`, для проведения априорного вывода, т.е. подсчета вероятности пропозициональной формулы на основе оценок вероятности исходного ФЗ, которые используют структуры `LinearForm` и `Extention` из более обширной структуры `ABN`, рассмотренной ранее. Метод `processKP` позволяет для каждой пропозициональной формулы из списка, содержащемся в исходном ФЗ, посчитать результат априорного вывода.

На Рис. 14 изображены взаимосвязи классов и интерфейсов, которые были описаны.

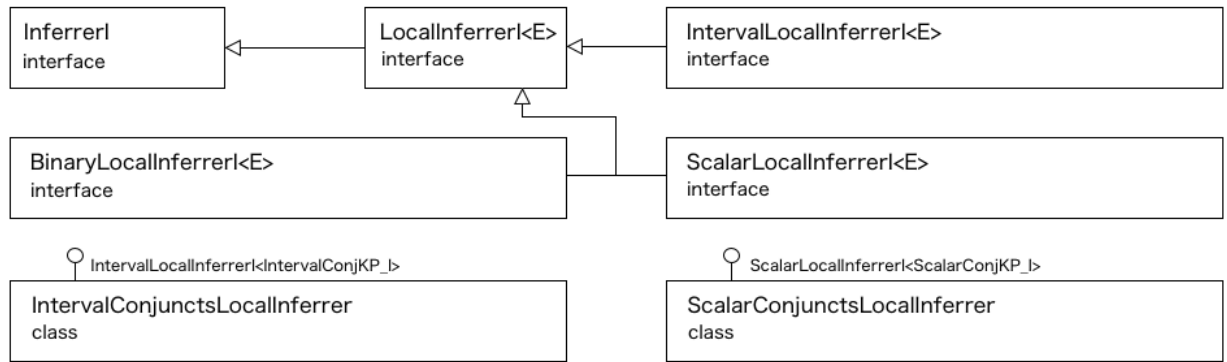


Рис. 14. Диаграмма структуры инфереров.

### 3.4.2. Структура Propagator апостериорного логико-вероятностного вывода

Структура Propagator, состоящая из классов, абстрактных классов и интерфейсов, отвечает за апостериорный логико-вероятностный вывод, а точнее за пропагацию детерминированного, стохастического или неточного свидетельства во ФЗ со скалярными или интервальными оценками вероятности. Далее приведен только срез структуры, соответствующий пропагации детерминированного свидетельства во ФЗ со скалярными оценками вероятности, для остальных случаев структуры аналогичные.

Интерфейс PropagatorI содержит один метод, которые позволяет узнать, какая ошибка произошла при работе программы.

Единственный метод интерфейса ScalarPropagatorI возвращает оценку свидетельства, пропагируемого во ФЗ. Оценка, полученная после пропагации свидетельства является решением первой задачи апостериорного логико-вероятностного вывода.

С помощью методов интерфейса LocalPropagatorI можно установить ФЗ, в котором будет производиться пропагация, выполнить пропагацию и получить пересчитанный ФЗ после пропагации.

Абстрактный класс ScalarConjunctsLocalPropagator наследует методы ранее описанных интерфейсов, а также задает новые методы, которые позволяют проверить ФЗ и свидетельство на «адекватность». Например обязательное правило для свидетельства то, что идеал конъюн-

юнктов, на котором оно построено, должно быть подмножеством идеала конъюнктов  $\Phi Z$ , в который оно пропагируется. Если это условие не выполняется, то пропация произойти не может.

Интерфейс `DeterministicScalarConjunctsLocalPropagatorI` представляет из себя обобщенный интерфейс, который наследует методы ранее приведенных интерфейсов.

Класс `DeterministicScalarConjunctsLocalPropagator` наследует от рассмотренного абстрактного класса `ScalarConjunctsLocalPropagator` и от интерфейса `DeterministicScalarConjunctsLocalPropagatorI` и определяет методы, которые ранее не были определены.

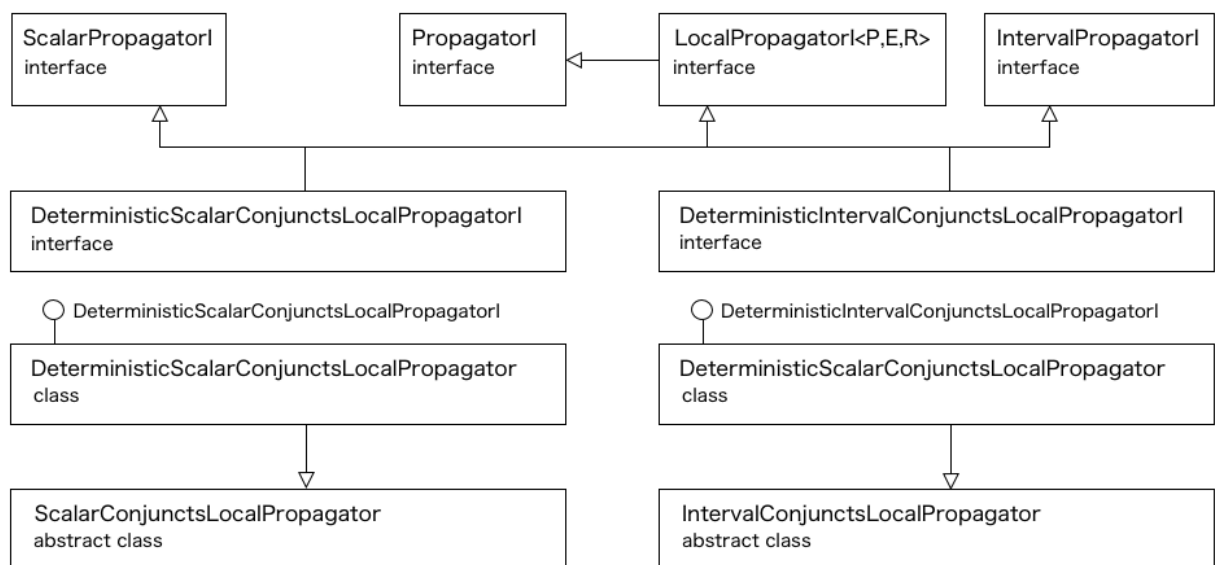


Рис. 15. Диаграмма структуры пропагаторов.

На Рис. 15 изображены взаимосвязи той структуры `Propagator`.

### 3.4.3. Вспомогательные структуры

В этом разделе приведем классы, используемые в основных структурах `ABN`, `Inferer` и `Propagator`.

Класс `LP` является классом-фасадом для импортируемой библиотеки `lp_solve55`, предназначенной для решения задач линейного программирования (ЛП). Этот класс унифицирует процесс решения задач ЛП, используемых при пропагации свидетельств, класс - прослойка, позволяющий задавать количество переменных, используемых в задаче ЛП,

коэффициенты у переменных в целевой функции и в ограничениях. Также имеется возможность решить задачу ЛП, как по максимизации целевой функции, так и по минимизации. В результате решения задачи мы получаем либо оптимальные значения рассматриваемых переменных, либо информацию о том, что задача не имеет решения. Также имеется возможность получения более подробной информации о возникшей проблеме. Все эти функции поддерживаются используемой библиотекой (`lp_solve55`) и к большинству из них был налажен доступ через вспомогательный класс `LP`.

Структура `Alphabet` позволяет работать с алфавитом, например с глобальными и локальными индексами атомов, находить строковое представление атома по его глобальному индексу. В структуре `Exceptions` содержатся классы, которые задают новые виды исключений. Пространство `Collectors` содержит классы, которые упрощают подсчет математического ожидания в пропагаторах. Структура `MatrixTransform` определяет множество операций с матрицами  $\mathbf{I}_n$  и  $\mathbf{J}_n$ . Класс `Utils` является вспомогательным. С помощью его методов можно, например сравнить на равенство два вещественных числа.

Вынесение некоторых функций и свойств в отдельные вспомогательные классы эффективно благодаря тому, что происходит «разделение ответственности» и в случае, если нам будет необходимо заменить какую-либо функцию, то мы сможем заменить ее всего лишь в одном месте.

### 3.5. Выводы по главе

В данной главе разобраны принципы наследования классов и интерфейсов в разработанной в рамках проекта системе. Рассмотрены структуры разработанного комплекса программ, реализующие создание ФЗ и осуществление локального логико-вероятностного вывода, а также те части комплекса, которые являются вспомогательными, для ранее упомянутых.

## 4. Программная реализация

### 4.1. Введение

Ранее были описаны структуры и их взаимосвязь в комплексе программ. В данной главе приведен основной спектр функциональности разработанного комплекса. Описаны примеры работы и некоторые математические выкладки для рассматриваемых примеров. Также в некоторых случаях приведены выдержки из исходного кода, которые реализуют соответствующую функциональность.

### 4.2. Примеры работы программного комплекса. Проверка и поддержание непротиворечивости

Рассмотрим пример проверки непротиворечивости и ее поддержание во фрагментах знаний. Поддержать непротиворечивость, то есть сузить оценки до непротиворечивых, можно только в интервальном ФЗ, в скалярном же мы можем только проверить, являются ли оценки согласованными или нет.

**Пример проверки непротиворечивости в скалярном ФЗ.** На вход поступает ФЗ  $(C, \mathbf{P}_c)$  со скалярными оценками вероятности

$$\mathbf{P}_c = \begin{pmatrix} 1 \\ 0.7 \\ 0.5 \\ 0.3 \end{pmatrix}.$$

В результате проверки непротиворечивости программа сообщает о непротиворечивости заданного ФЗ.

На листинге 1 приведен исходный код, который позволяет создать ФЗ scKP со скалярными оценками вероятности, построенный над идеалом конъюнктов, создать соответствующую ему машину для априорного вывода `inf_sc` и с помощью метода `processKP()` провести проверку



непротиворечивости . Узнать, согласован данный ФЗ или нет можно узнать с помощью метода `isConsistent()`.

---

```
ScalarConjKP_I scKP = new ScalarConjKP(
    Convert.ToInt64("'11'", 2),
    new[] {1, 0.7, 0.5, 0.3});
var inf_sc = new ScalarConjunctsLocalInferer();
inf_sc.processKP(scKP);
inf_sc.isConsistent();
```

---

Листинг 1. Проверка непротиворечивости в скалярном ФЗ

**Пример проверки непротиворечивости в интервальном ФЗ.** Фрагмент знаний  $(C, \mathbf{P}_c^-, \mathbf{P}_c^+)$  с интервальными оценками вероятности.

$$\mathbf{P}_c^- = \begin{pmatrix} 1 \\ 0.46 \\ 0.29 \\ 0.33 \end{pmatrix}$$

и

$$\mathbf{P}_c^+ = \begin{pmatrix} 1 \\ 0.7 \\ 0.5 \\ 0.9 \end{pmatrix}.$$

В результате программа выдаст суженные оценки вероятности:

$$\mathbf{P}_c^- = \begin{pmatrix} 1 \\ 0.46 \\ 0.33 \\ 0.33 \end{pmatrix}$$

и

$$\mathbf{P}_c^+ = \begin{pmatrix} 1 \\ 0.7 \\ 0.5 \\ 0.5 \end{pmatrix}.$$

а также сообщит о том, что оценки вероятности заданного ФЗ согласованы.

На листинге 2 приведен исходный код, который позволяет создать ФЗ `intKP` с интервальными оценками вероятности, построенный над идеалом конъюнктов, создать соответствующую ему машину для априорного вывода `inf` и с помощью метода `processKP()` провести проверку непротиворечивости. Узнать, согласован данный ФЗ или нет можно узнать с помощью метода `isConsistent()`. В дальнейшем не будем останавливаться на создании ФЗ подробно.

---

```
IntervalConjKP_I intKP = new IntervalConjKP(
    Convert.ToInt64("'11'", 2),
    new[] {1, 0.46, 0.29, 0.33},
    new[] {1, 0.7, 0.5, 0.9});
Console.WriteLine(intKP.ToString());
var inf = new IntervalConjunctsLocalInferer();
inf.processKP(intKP);
inf.isConsistent();
```

---

Листинг 2. Проверка непротиворечивости в интервальном ФЗ

### 4.3. Примеры работы программного комплекса. Априорный вывод

**Пример априорного вывода во ФЗ со скалярными оценками.** На вход подается ФЗ  $(C, P_c)$ , заданный над алфавитом мощности 2, т.е. состоящий из двух атомов  $A = \{x_1, x_2\}$ , со скалярными оценками вероятности:

$$P_c = \begin{pmatrix} 1 \\ 0.7 \\ 0.5 \\ 0.3 \end{pmatrix}.$$

Также мы хотим посчитать вероятность пропозициональной формулы  $x_1 \implies x_2$  на основе оценок вероятностей данного ФЗ и вектора коэффициентов, которые соответствуют коэффициентам в разложении данной пропозициональной формулы в СДНФ по конъюнктам из ФЗ:

$$\mathbf{P}_c^a = \begin{pmatrix} 1 \\ -1 \\ 0 \\ 1 \end{pmatrix}.$$

Вычисляется вероятность пропозициональной формулы:  $p(x_1 \implies x_2) = 0.6$ . Также есть возможность узнать, можно ли сузить данные оценки вероятности и совместны ли они. Программа сообщает, что оценки совместные, но сузить их нельзя. Это следует из того, что оценки точечные.

На листинге 3 с помощью метода `addLinearForm()` задается пропозициональная формула для уже существующего скалярного ФЗ, создание которого описано на листинге 1. С помощью метода `inf_sc.processKP()` проводится априорный вывод и считается нижняя и верхняя оценка вероятность поступившей формулы. В данном случае границы совпадут.

---

```
scKP.addLinearForm('Implication', 1, -1, 0, 1);
inf_sc.processKP(scKP);
inf_sc.getResult().getLinearForm('Implication').getLB();
inf_sc.getResult().getLinearForm('Implication').getUB();
```

---

Листинг 3. Априорный вывод в ФЗ со скалярными оценками

### Пример априорного вывода во ФЗ с интервальными оценками.

На вход подается ФЗ  $(C, \mathbf{P}_c^-, \mathbf{P}_c^+)$ , заданный над алфавитом мощности 2, т.е. состоящий из двух атомов  $A = \{x_1, x_2\}$ , с интервальными оценками вероятности:

$$\mathbf{P}_c^- = \begin{pmatrix} 1 \\ 0.46 \\ 0.29 \\ 0.33 \end{pmatrix}$$

и

$$\mathbf{P}_c^+ = \begin{pmatrix} 1 \\ 0.7 \\ 0.5 \\ 0.9 \end{pmatrix}.$$

Посчитаем вероятность пропозициональной ф-лы  $x_1 \equiv x_2$  на основе оценок вероятностей данного ФЗ и вектора коэффициентов, которые соответствуют коэффициентам в разложении данной пропозициональной формулы в СДНФ:

$$\mathbf{P}_c^a = \begin{pmatrix} 1 \\ -1 \\ -1 \\ 2 \end{pmatrix}.$$

В результате проверки на непротиворечивость оценки уточнятся:

$$\mathbf{P}_c^- = \begin{pmatrix} 1 \\ 0.46 \\ 0.33 \\ 0.33 \end{pmatrix}$$

и

$$\mathbf{P}_c^+ = \begin{pmatrix} 1 \\ 0.7 \\ 0.5 \\ 0.5 \end{pmatrix}.$$

В данном случае мы получаем интервальные оценки вероятности пропозициональной формулы:  $0.46 \leq p(x_1 \equiv x_2) \leq 1$ . Программа сообщает о том, что совместные и могут быть сужены.

На листинге 4 приведен исходный код решения задачи априорного вывода для интервального ФЗ. Вычисление происходит аналогично вычислению на листинге 3.

---

```
intKP.addLinearForm("Equivalence", 1, -1, -1, 2);
inf.processKP(intKP);
inf.getResult().getLinearForm("Equivalence").getLB();
```

```
inf.getResult().getLinearForm('Equivalence').getUB();
```

---

Листинг 4. Априорный вывод в ФЗ с интервальными оценками

## 4.4. Примеры работы программного комплекса. Апостериорный вывод

Рассмотрим примеры пропагации различных видов свидетельств в интервальные или скалярные ФЗ.

**Пример пропагации детерминированного свидетельства во фрагмент знаний со скалярными оценками.** На вход подается ФЗ  $(C, \mathbf{P}_c)$  со скалярными оценками вероятности

$$\mathbf{P}_c = \begin{pmatrix} 1 \\ 0.7 \\ 0.5 \\ 0.3 \end{pmatrix}$$

и детерминированное свидетельство :  $\langle x_1 \rangle$ .

В результате пропагации мы получаем в качестве решения первой задачи апостериорного вывода значение  $p(\langle x_1 \rangle | (C, \mathbf{P}_c)) = 0.7$ , а в качестве решения второй задачи апостериорного вывода вектор с оценками вероятности:

$$\mathbf{P}_c^{\langle x_1 \rangle} = \begin{pmatrix} 1 \\ 1 \\ 0.4285 \\ 0.4285 \end{pmatrix}.$$

На листинге 5 приведен исходный код решения задач апостериорного вывода для скалярного ФЗ и детерминированного свидетельства. Сначала создается соответствующий ФЗ `kr`, свидетельство `evid` и машина апостериорного вывода `scrpor`, осуществляющая решение задач.

С помощью метода `propagate()` осуществляется решение задач апостериорного вывода. С помощью метода `getEvidenceProbability()` можно получить решение первой задачи, а с помощью `getResult()` – второй.

---

```

ScalarConjKP_I kp = new ScalarConjKP(
    Convert.ToInt64("11", 2),
    new[] {1, 0.7, 0.5, 0.3});
DeterministicScalarConjunctsLocalPropagatorI scprop = new
    DeterministicScalarConjunctsLocalPropagator(kp);
BinaryConjKP_I evid = new BinaryConjKP(
    Convert.ToInt64("01", 2));
scprop.propagate(evid);
scprop.getEvidenceProbability();
scprop.getResult().GetPointEstimate();

```

---

Листинг 5. Пропагация детерминированного свидетельства во фрагмент знаний со скалярными оценками

**Пример пропагации детерминированного свидетельства в ФЗ с интервальными оценками.** На вход подается ФЗ  $(C, \mathbf{P}_c^-, \mathbf{P}_c^+)$  с интервальными оценками вероятности:

$$\mathbf{P}_c^- = \begin{pmatrix} 1 \\ 0.2 \\ 0.3 \\ 0.06 \end{pmatrix}$$

и

$$\mathbf{P}_c^+ = \begin{pmatrix} 1 \\ 0.4 \\ 0.58 \\ 0.29 \end{pmatrix}$$

и детерминированное свидетельство :  $\langle x_1 \rangle$ .

В результате пропагации мы получаем в качестве решения первой задачи апостериорного вывода  $0.2 \leq p(\langle x_1 \rangle | (C, \mathbf{P}_c^-, \mathbf{P}_c^+)) \leq 0.4$ , а в качестве решения второй задачи апостериорного вывода векторы с оценками вероятности:

$$\mathbf{P}_c^{\langle x_1 \rangle, -} = \begin{pmatrix} 1 \\ 1 \\ 0.15 \\ 0.15 \end{pmatrix}$$

и

$$\mathbf{P}_c^{\langle x_1 \rangle, -} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

На листинге 6 приведен исходный код решения задач апостериорного вывода для интервального ФЗ и детерминированного свидетельства. Вычисление и получение результата происходит аналогично листингу 5.

---

```
IntervalConjKP_I kp = new IntervalConjKP(
    Convert.ToInt64("11", 2),
    new[] {1, 0.2, 0.3, 0.06},
    new[] {1, 0.4, 0.58, 0.29});
DeterministicIntervalConjunctsLocalPropagator prop = new
    DeterministicIntervalConjunctsLocalPropagator(kp);
var evid = new BinaryConjKP(
    Convert.ToInt64("01", 2));
prop.propagate(evid);

prop.getEvidenceProbabilityLB();
prop.getEvidenceProbabilityUB();

prop.getResult().GetBoundsL();
prop.getResult().GetBoundsU();
```

---

Листинг 6. Пропагация детерминированного свидетельства в ФЗ с интервальными оценками

**Пример пропагации стохастического свидетельства во фрагмент знаний со скалярными оценками.** На вход подается ФЗ  $(C, \mathbf{P}_c)$  со скалярными оценками вероятности

$$\mathbf{P}_c = \begin{pmatrix} 1 \\ 0.6 \\ 0.4 \\ 0.2 \end{pmatrix}$$

и стохастическое свидетельство :  $\langle\langle C', \mathbf{P}_c^a \rangle\rangle$ , заданное над алфавитом  $A_{C'} = \{x_1\}$ .

$$\mathbf{P}_c^a = \begin{pmatrix} 1 \\ 0.3 \end{pmatrix}.$$

В результате пропагации мы получаем в качестве решения первой задачи апостериорного вывода :  $p(\langle\langle C', \mathbf{P}_c^a \rangle\rangle | (C, \mathbf{P}_c)) = 0.46$ , а в качестве решения второй задачи апостериорного вывода вектор с оценками вероятности:

$$\mathbf{P}_c^{\langle\langle C', \mathbf{P}_c^a \rangle\rangle} = \begin{pmatrix} 1 \\ 0.3 \\ 0.45 \\ 0.1 \end{pmatrix}.$$

На листинге 7 приведен исходный код решения задач апостериорного вывода для скалярного ФЗ и стохастического свидетельства. Вычисление и получение результата происходит аналогично листингу 5.

---

```

ScalarConjKP_I kp = new ScalarConjKP(
    Convert.ToInt64("11", 2),
    new[] {1, 0.6, 0.4, 0.2});
StochasticScalarConjunctsLocalPropagatorI scprop = new
    StochasticScalarConjunctsLocalPropagator(kp);
ScalarConjKP_I evid = new ScalarConjKP(
    Convert.ToInt64("01", 2),
    new[] {1, 0.3});
scprop.propagate(evid);
scprop.getEvidenceProbability();
scprop.getResult().GetPointEstimate();

```

---

Листинг 7. Пропагация стохастического свидетельства во фрагмент знаний со скалярными оценками



**Пример пропагации стохастического свидетельства во фрагмент знаний с интервальными оценками.** На вход подается ФЗ  $(C, \mathbf{P}_c^-, \mathbf{P}_c^+)$  с интервальными оценками вероятности

$$\mathbf{P}_c^- = \begin{pmatrix} 1 \\ 0.2 \\ 0.3 \\ 0.06 \end{pmatrix}$$

и

$$\mathbf{P}_c^+ = \begin{pmatrix} 1 \\ 0.4 \\ 0.58 \\ 0.29 \end{pmatrix}$$

и стохастическое свидетельство :  $\langle (C', \mathbf{P}_c^a) \rangle$ , заданное над алфавитом  $A_{C'} = \{x_1\}$ .

$$\mathbf{P}_c^a = \begin{pmatrix} 1 \\ 0.3 \end{pmatrix}.$$

В результате пропагации мы получаем в качестве решения первой задачи апостериорного вывода :  $0.48 \leq p(\langle (C', \mathbf{P}_c^a) \rangle | (C, \mathbf{P}_c^-, \mathbf{P}_c^+)) \leq 0.68$ , а в качестве решения второй задачи апостериорного вывода векторы с оценками вероятности:

$$\mathbf{P}_c^{\langle (C', \mathbf{P}_c^a) \rangle, -} = \begin{pmatrix} 1 \\ 0.3 \\ 0.05485 \\ 0.045 \end{pmatrix}$$

и

$$\mathbf{P}_c^{\langle (C', \mathbf{P}_c^a) \rangle, +} = \begin{pmatrix} 1 \\ 0.3 \\ 0.9066 \\ 0.3 \end{pmatrix}.$$

На листинге 8 приведен исходный код решения задач апостериорно-

го вывода для интервального ФЗ и стохастического свидетельства. Вычисление и получение результата происходит аналогично листингу 5.

---

```
IntervalConjKP_I kp = new IntervalConjKP(
    Convert.ToInt64("11", 2),
    new[] {1, 0.2, 0.3, 0.06},
    new[] {1, 0.4, 0.58, 0.29});
StochasticIntervalConjunctsLocalPropagatorI scprop = new
    StochasticIntervalConjunctsLocalPropagator(kp);
ScalarConjKP_I evid = new ScalarConjKP(
    Convert.ToInt64("01", 2),
    new[] {1, 0.3});
scprop.propagate(evid);

scprop.getEvidenceProbabilityLB();
scprop.getEvidenceProbabilityUB();

scprop.getResult().GetBoundsL();
scprop.getResult().GetBoundsU();
```

---

Листинг 8. Пропагация стохастического свидетельства во фрагмент знаний с интервальными оценками

**Пример пропагации неточного свидетельства во фрагмент знаний со скалярными оценками.** На вход подается ФЗ  $(C, \mathbf{P}_c)$  со скалярными оценками вероятности

$$\mathbf{P}_c = \begin{pmatrix} 1 \\ 0.6 \\ 0.4 \\ 0.2 \end{pmatrix}$$

и неточное свидетельство :  $\langle (C', \mathbf{P}_c^{a,-}, \mathbf{P}_c^{a,+}) \rangle$ , заданное над алфавитом  $A_{C'} = \{x_1\}$ .

$$\mathbf{P}_c^{a,-} = \begin{pmatrix} 1 \\ 0.1 \end{pmatrix}$$

и

$$\mathbf{P}_c^{a,+} = \begin{pmatrix} 1 \\ 0.6 \end{pmatrix}.$$

В результате пропагации мы получаем в качестве решения первой

задачи апостериорного вывода :  $0.42 \leq p(\langle(C', \mathbf{P}_c^{a,-}, \mathbf{P}_c^{a,+})\rangle | (C, \mathbf{P}_c)) \leq 0.52$ , а в качестве решения второй задачи апостериорного вывода вектор с оценками вероятности:

$$\mathbf{P}_c^{\langle(C', \mathbf{P}_c^{a,-}, \mathbf{P}_c^{a,+})\rangle, -} = \begin{pmatrix} 1 \\ 0.1 \\ 0.4 \\ 0.0333 \end{pmatrix}$$

и

$$\mathbf{P}_c^{\langle(C', \mathbf{P}_c^{a,-}, \mathbf{P}_c^{a,+})\rangle, +} = \begin{pmatrix} 1 \\ 0.6 \\ 0.48333 \\ 0.2 \end{pmatrix}.$$

На листинге 9 приведен исходный код решения задач апостериорного вывода для скалярного ФЗ и неточного свидетельства. Вычисление и получение результата происходит аналогично листингу 5.

---

```

ScalarConjKP_I kp = new ScalarConjKP(
    Convert.ToInt64("11", 2),
    new[] {1, 0.6, 0.4, 0.2});
ImpreciseScalarConjunctsLocalPropagatorI scprop = new
    ImpreciseScalarConjunctsLocalPropagator();
scprop.setPattern(kp);
IntervalConjKP_I evid = new IntervalConjKP(
    Convert.ToInt64("01", 2),
    new[] {1, 0.1},
    new[] {1, 0.6});
scprop.propagate(evid);

scprop.getEvidenceProbabilityLB();
scprop.getEvidenceProbabilityUB();

scprop.getResult().GetBoundsL();
scprop.getResult().GetBoundsU();

```

---

Листинг 9. Пропагация неточного свидетельства во фрагмент знаний со скалярными оценками

**Пример пропагации неточного свидетельства во фрагмент знаний с интервальными оценками.** На вход подается ФЗ  $(C, \mathbf{P}_c^-, \mathbf{P}_c^+)$  с интервальными оценками вероятности

$$\mathbf{P}_c^- = \begin{pmatrix} 1 \\ 0.2 \\ 0.5 \\ 0.06 \end{pmatrix}$$

и

$$\mathbf{P}_c^+ = \begin{pmatrix} 1 \\ 0.7 \\ 0.87 \\ 0.4 \end{pmatrix}$$

и неточное свидетельство :  $\langle (C', \mathbf{P}_c^{a,-}, \mathbf{P}_c^{a,+}) \rangle$ , заданное над алфавитом  $A_{C'} = \{x_1\}$ .

$$\mathbf{P}_c^{a,-} = \begin{pmatrix} 1 \\ 0.2 \end{pmatrix}$$

и

$$\mathbf{P}_c^{a,+} = \begin{pmatrix} 1 \\ 0.73 \end{pmatrix}.$$

В результате пропагации мы получаем в качестве решения первой задачи апостериорного вывода :  $0.227 \leq p(\langle (C', \mathbf{P}_c^a) \rangle | (C, \mathbf{P}_c^-, \mathbf{P}_c^+)) \leq 0.78$ , а в качестве решения второй задачи апостериорного вывода векторы с оценками вероятности:

$$\mathbf{P}_c^{\langle (C', \mathbf{P}_c^a) \rangle, -} = \begin{pmatrix} 1 \\ 0.2 \\ 0.12321 \\ 0.021428 \end{pmatrix}$$

и

$$\mathbf{P}_c^{\langle(C', \mathbf{P}_c^a)\rangle, +} = \begin{pmatrix} 1 \\ 0.73 \\ 1 \\ 0.73 \end{pmatrix}.$$

На листинге 10 приведен исходный код решения задач апостериорного вывода для интервального ФЗ и неточного свидетельства. Вычисление и получение результата происходит аналогично листингу 5.

---

```
IntervalConjKP_I kp = new IntervalConjKP(
    Convert.ToInt64("11", 2),
    new[] {1, 0.2, 0.5, 0.06},
    new[] {1, 0.7, 0.87, 0.4});
ImpreciseIntervalConjunctsLocalPropagatorI scprop = new
    ImpreciseIntervalConjunctsLocalPropagator();
scprop.setPattern(kp);
IntervalConjKP_I evid = new IntervalConjKP(
    Convert.ToInt64("01", 2),
    new[] {1, 0.2},
    new[] {1, 0.73});
scprop.propagate(evid);

scprop.getEvidenceProbabilityLB();
scprop.getEvidenceProbabilityUB();

scprop.getResult().GetBoundsL();
scprop.getResult().GetBoundsU();
```

---

Листинг 10. Пропагация неточного свидетельства во фрагмент знаний с интервальными оценкам

## 4.5. Выводы по главе

Функциональность разработанного программного комплекса покрывает базовую функциональность, необходимую для осуществления локального логико-вероятностного вывода над фрагментами знаний как со скалярными оценками вероятности, так и с интервальными. В некоторых случаях приведен исходный код, показывающий каким образом создается ФЗ, как для него задаются оценки и выполняются основные

операции.

# Заключение

В ходе работы над выпускной квалификационной работой бакалавра была достигнута основная цель – реализована библиотека на C#, производящая локальный логико-вероятностный вывод в АБС. Она состоит из структуры, создающей ФЗ и свидетельства, построенные над разными множествами, с различными типами оценок вероятности, машин вывода, осуществляющих решение задач априорного и апостериорного выводов и вспомогательных классов.

Все задачи были выполнены, получены следующие результаты:

- доказаны две теоремы, формализующие задачи апостериорный вывод для фрагментов знаний, заданных над идеалом дизъюнктов;
- реализованы ФЗ и машины вывода на языке C#;
- разработаны примеры и документация.

Результаты вошли в 4 публикации:

1. Золотин А.А., Мальчевская Е.А. Матрично-векторные алгоритмы локального апостериорного вывода в алгебраических байесовских сетях над идеалами дизъюнктов // Материалы международной конференции по мягким вычислениям и измерениям (в печати)
2. Zolotin A.A., Malchevskaya E.A. Matrix-Vector Algorithms of Local Posteriori Inference in Algebraic Bayesian Networks on Ideal of Disjuncts // International Conference on Soft Computing and Measurements (in press)
3. Mal'chevskaya E.A., Berezin A.I., Zolotin A.A., Tulupyev A.L. Algebraic Bayesian Networks: Local Probabilistic-Logic Inference Machine Architecture and Set of Minimal Joint Graphs // 1st International Scientific Conference «Intelligent information technologies for industry» (in press)

4. Мальчевская Е.А., Золотин А.А. Логико-вероятностный вывод в АБС: архитектура и примеры использования программного комплекса на языке C# // 3-я Всероссийская Пospelовская конференция с международным участием «Гибридные и синергетические интеллектуальные системы» (в печати)

Результаты были представлены на двух конференциях: 1-ой Международной научной конференции «Интеллектуальные информационные технологии в технике и на производстве» и Всероссийской научной конференции по проблемам информатики СПИСОК-2016.

Была подана заявка на регистрацию программы для ЭВМ: Мальчевская Е.А., Тулупьев А.Л. Система представления фрагментов знаний алгебраических байесовских сетей Algebraic Bayesian Networks Knowledge Patterns Modeler, Version 01 for CSharp (AlgBN KP Modeler cs.v.01).

Работа выполнялась под руководством А.Л. Тулупьева, на базе лаборатории теоретических и междисциплинарных проблем информатики СПИИРАН; кроме того, разработки были частично поддержаны грантами РФФИ 15-01-09001-а — «Комбинированный логико-вероятностный графический подход к представлению и обработке систем знаний с неопределенностью: алгебраические байесовские сети и родственные модели», 12-01-00945-а — «Развитие теории алгебраических байесовских сетей и родственных им логико-вероятностных графических моделей систем знаний с неопределенностью», 09-01-00861-а — «Методология построения интеллектуальных систем поддержки принятия решений на основе баз фрагментов знаний с вероятностной неопределенностью», а также проектом по ФЦНТП «Исследования и разработки по приоритетным направлениям развития науки и техники на 2002–2006 годы», 2006-РИ-19.0/001/211, Государственный контракт от «28» февраля 2006 г., № 02.442.11.7289, «Направленные циклы в байесовских сетях доверия: вероятностная семантика и алгоритмы логико-вероятностного вывода для программных комплексов с байесовской интеллектуальной компонентой».

Полученные результаты позволяют в будущем развить функциональность данной библиотеки, реализовывать глобальный логико-веро-



ятностный вывод, а также интегрироваться с другими подпроектами, которые связаны со структурной компонентой библиотеки.

## Список литературы

- [1] Золотин А.А., Мальчевская Е.А. Матрично-векторные алгоритмы локального апостериорного вывода в алгебраических байесовских сетях над идеалами дизъюнктов // Материалы международной конференции по мягким вычислениям и измерениям (в печати)
- [2] Зотов М.А., Тулупьев А.Л. Вторичная структура алгебраических байесовских сетей: статистическая оценка сложности прямого алгоритма синтеза // Международная конференция по мягким вычислениям и измерениям. 2015. Т. 2. № Секции 4–7. С. 158–162
- [3] Сироткин А.В. Алгебраические байесовские сети: вычислительная сложность алгоритмов логико-вероятностного вывода в условиях неопределенности. — Изд-во С.-Петербур. ун-та, 2011.
- [4] Сироткин А.В. Вычислительная сложность локального апостериорного вывода в алгебраических байесовских сетях// Труды СПИИРАН. 2011. Вып.3(18). С.188-214.
- [5] Сироткин А.В. Локальный априорный вывод в алгебраических байесовских сетях: комплекс основных алгоритмов // Труды СПИИРАН. Вып.5. СПб.: Наука, 2007. С.100-111.
- [6] Сироткин А.В. Проверка и поддержание непротиворечивости алгебраических байесовских сетей: вычислительная сложность алгоритмов // Труды СПИИРАН. 2010. Вып.4(15). С.162-192.
- [7] Сироткин А.В., Тулупьев А.Л. Матричные уравнения локального логико-вероятностного вывода в алгебраических байесовских сетях // Труды СПИИРАН. Вып.6. СПб.: Наука, 2008. С.134-143.
- [8] Тулупьев А.Л. Алгебраические байесовские сети: глобальный логико-вероятностный вывод в деревьях смежности: Учеб. пособие. СПб.: СПбГУ; ООО Издательство «Анатолия», 2007. 40 с. (Сер. Элементы мягких вычислений).

- [9] Тулупьев А.Л. Алгебраические байесовские сети: логико-вероятностная графическая модель баз фрагментов знаний с неопределенностью. Санкт-Петербургский институт информатики и автоматизации РАН', — Изд-во С.-Петерб. ун-та, 2009.
- [10] Тулупьев А.Л. Алгебраические байесовские сети: локальный логико-вероятностный вывод: Учеб. пособие. СПб.: СПбГУ; ООО Издательство «Анатолия», 2007. 80 с. (Сер. Элементы мягких вычислений).
- [11] Тулупьев А.Л. Алгебраические байесовские сети: локальный логико-вероятностный вывод: Учеб. пособие. СПб.: СПбГУ; ООО Издательство «Анатолия», 2007. 80 с. (Сер. Элементы мягких вычислений).
- [12] Тулупьев А.Л. Алгебраические байесовские сети: система операций глобального логико-вероятностного вывода // Информационно-измерительные и управляющие системы. 2010. №11. С. 65–72.
- [13] Тулупьев А.Л. Апостериорные оценки вероятностей в алгебраических байесовских сетях // Вестн. С.-Петерб. ун-та. Сер. 10. 2012. Вып. 2. С. 51–59.
- [14] Тулупьев А.Л. Байесовские сети: логико-вероятностный вывод в циклах. СПб.: Изд-во С.-Петербургского ун-та, 2008. 140 с. (Сер. Элементы мягких вычислений.)
- [15] Тулупьев А.Л. Вероятностная логика и вероятностные графические модели в базах фрагментов знаний с неопределенностью // Интегрированные модели, мягкие вычисления, вероятностные системы и комплексы программ в искусственном интеллекте. Научно-практическая конференция студентов, аспирантов, молодых ученых и специалистов (Коломна, 26–27 мая 2009 г.). Научные доклады. В 2-х т. Т. 1. М.: Физматлит, 2009. С. 26–46. Наука, 2006. С. 198–227.
- [16] Тулупьев А.Л. Дерево смежности с идеалами конъюнктов как

ациклическая алгебраическая байесовская сеть // Тр. СПИИРАН. 2006. Т. 1. Вып. 3. С. 198–227.

- [17] Тулупьев А.Л. Система для апостериорного вывода в алгебраических байесовских сетях и их фрагментах Algebraic Bayesian Networks Propagator, Version 01 for Java (AlgBN Propagator j.v.01) (Свидетельство). Свид. о гос. рег. progr. для ЭВМ. Рег. № 2009613804 (16.07.2009). Роспатент.. Бюлл. «Прогр. для ЭВМ, БД, топол. инт. микросх.». 2009. № 4. С. 65.
- [18] Тулупьев А.Л. Система для синтеза непротиворечивых алгебраических байесовских сетей и их фрагментов Algebraic Bayesian Networks Inferer, Version 01 for Java (AlgBN Inferer j.v.01) (Свидетельство). Свид. о гос. рег. progr. для ЭВМ. Рег. № 2009613803 (16.07.2009). Роспатент.. Бюлл. «Прогр. для ЭВМ, БД, топол. инт. микросх.». 2009. № 4. С. 65.
- [19] Тулупьев А.Л. Система представления алгебраических байесовских сетей и их фрагментов Algebraic Bayesian Networks Modeler, Version 01 for Java (AlgBN Modeler j.v.01) (Свидетельство). Свид. о гос. рег. progr. для ЭВМ. Рег. № 2009613802 (16.07.2009). Роспатент.. Бюлл. «Прогр. для ЭВМ, БД, топол. инт. микросх.». 2009. № 4. С. 64–65.
- [20] Тулупьев А.Л., Николенко С.И., Сироткин А.В. Байесовские сети: логико-вероятностный подход. СПб.: Наука, 2006. 607 с.
- [21] Тулупьев А.Л., Сироткин А.В. Программа для моделирования фрагмента знаний алгебраической байесовской сети, поддержания его непротиворечивости и апостериорного вывода в нем Algebraic Bayesian Network Knowledge Pattern Modeler, Reconciler and Propagator Version 01 for C++ (AlgBN KP MRP cpp.v.01) (Свидетельство). Свид. о гос. регистрации программы для ЭВМ. Рег. № 2010615242(13.08.2010). Роспатент.

- [22] Тулупьев А.Л., Сироткин А.В., Золотин А.А. Матричные уравнения нормирующих множителей в локальном апостериорном выводе оценок истинности в алгебраических байесовских сетях // Вестник СПбГУ. Сер. 1. Т. 2 (60). 2015. Вып. 3. С. 379–386.
- [23] Тулупьев А.Л., Сироткин А.В., Николенко С.И. Байесовские сети доверия: логико-вероятностный вывод в ациклических направленных графах. СПб.: Изд-во С.-Петерб. ун-та, 2009. 400 с.
- [24] Фильченков А.А., Тулупьев А.Л. Связность и ацикличность первичной структуры алгебраической байесовской сети // Вестник СПбГУ. Сер. 1. 2013. С. 110–119.